FACULTY OF COMPUTER SCIENCE AND
INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
SESSION 2004/2005

JAPANESE COMPUTER-AIDED LEARNING PACKAGE

SITI NUR HIDAYAH JOHAR (WEK 020231)

Under the Supervision of
Assoc Prof Dr Ow Siew Hock

Moderator
Ms Su Moon Ting

This project is submitted to the Faculty of Science and Information Technology, University of Malaya,
in partial fulfillment of the requirement of the Bachelor of Computer Science

Abstract

NihonGo, a Japanese learning software, is developed to teach the basic Japanese language to the public in an easy way. It comprises 8 modules. They are Maintenance, Writing, Lessons, Exercises, Quizzes, Games, Karaoke and Dictionary. It is a bilingual system that provides text-to-speech synthesizers (English, Malay and Japanese) and instructions in either English or Malay.

The NihonGo's administrator can updates and add new terms, or delete existing terms in Maintenance module. Learners choose the topics to study in Lessons module. As to make sure they understand Japanese language grammars, they have to do exercises and quizzes. Games and Karaoke provides learning in a fun way and keep the learners relax. Dictionary allows the learners to search the definition of a Japanese word term.

NihonGo was developed using incremental evolution through rapid prototyping technique. The development tools used include Visual Basic.NET, Microsoft Access 2003, Adobe Photoshop CS, and Macromedia Flash MX 2004 and so on.

The NihonGo package can be used as a learning tool in schools. It is beneficial for organizations that want to instill learning Japanese language in an easy and fun way. Therefore, it also promotes the Japanese culture. Through interesting multimedia, NihonGo can attract the learners and thus enhance their learning processes.

Acknowledgement

First and foremost, I would to extend my heartfelt gratitude to my supervisor, Assoc. Prof. Dr. Ow Siew Hock who has, patiently, given many helpful and constructive suggestions, and thoughtful contribution throughout this project. Through her clear guidance, advices, Japanese language lessons and full freedom in developing this project, she has given me the opportunity to develop a system in full confidence and learn Japanese language.

Secondly, I would like to express my sincere appreciation to Ms. Su Moon Ting for sparing her precious time to be my moderator. Her constructive comments and feedback were highly regarded as an enhancement for my project.

My sincere thanks also go to the staffs of the Faculty of Computer Science and Information Technology of University Malaya for their assistance and co-operation for readying helps.

I am also most appreciative of the boundless help in sharing knowledge, provocative questions and suggestions by my course mate, Mr. Lee Boon Why, Mr Chong Soon Sang and Ms Quek Seow Peng to accomplish this project.

Finally, as always, I thank my family members and friends for their support, and guidance throughout the project development. I am greatly indebted and grateful to all of them.

Table of Contents

Abstract ii

Acknowledgement iii

Table of Contents iv

List of Tables ix

List of Figures x

List of Abbreviations xii

Chapter 1 Introduction 1

1.1 Project Initiation 1

1.2 Project Objective 1

1.3 Project Scopes 3

1.4 Project Outcome 3

1.5 Project Schedule 4

Chapter 2 Literature Review 6

2.1 Introduction to Literature Review 6

2.2 Computer Aided Learning (CAL) 6

2.3 Japanese

2.3.1 About Japanese Writing 9

2.3.2 Hiragana 9

2.3.3 Katakana 10

2.3.4 Kanji 10

2.4 Multimedia 12

2.5 A Survey On Current Learning Package 13

2.6 Summary of Reviewed Learning Package 18

2.7 A Survey On Seniors Project 19

2.8 Consideration of Programming Language

2.8.1 Visual Basic .NET 21

2.8.2 Visual Basic 6.0 23

2.8.3 Comparison between Visual Basic .NET (.NET) and Visual Basic 6.0 (VB6) 24

2.9 Consideration of Database

2.9.1 Microsoft Access 2003 25

2.9.2 Microsoft SQL Server 2000 27

2.9.3 Comparison Between Microsoft Access XP and Microsoft Server 2003 29

2.10 Consideration of Multimedia Tools

2.10.1 Macromedia Flash MX 30

2.10.2 Macromedia Director MX 32

2.10.3 Comparison Between Macromedia Flash MX and Macromedia Director MX 34

Chapter 3 System Analysis **36**

3.1 Introduction 36

3.2 Incremental Evolution through Rapid Prototyping 37

3.3 Advantages in using the Evolution through Rapid Prototyping 38

3.4 Research from Interview 39

3.5 Research from Internet 41

3.6 Research from Books 41

3.7 Research from the market 41

3.8 Research from articles and newspapers 42

3.9 Why Use Visual Basic.NET 42

3.10 Why Use Microsoft Access 43

3.11 Why Use Macromedia Flash 44

3.12 Why Use Adobe Photoshop 45

Chapter 4 System Analysis 46

4.1 Functional Requirement 46

4.2 Non-Functional Requirement 48

4.3 Software Requirement 50

4.4 Hardware Requirement 51

Chapter 5 System Design 52

5.1 Architectural Design 52

5.2 Process Model 58

5.3 Database Design 64

5.3.1 Physical Database Design 65

5.3.2 Logical Database Design 68

5.4 User Interface Design 68

Chapter 6 System Implementation **78**

6.1 Introduction 78

6.2 Development Environment 78

6.2.1 Software Requirements 78

6.2.2 Hardware Requirements 79

6.3 System Development 79

6.3.1 Database Development 79

6.3.2 User Interface Development 80

6.3.3 Application Development 80

6.4 System Documentation 82

Chapter 7 System Testing and Debugging **83**

7.1 Introduction 83

7.2 Testing Techniques Practiced 84

7.2.1 Ad-hoc testing 84

7.2.2 Black box testing 84

7.2.3 White Box testing 85

7.3 Testing the system 86

7.3.1 Unit testing 87

7.3.2 Integration testing 87

7.3.3 System testing 88

Chapter 8 System Evaluation and Conclusion **90**

8.1 Problem Encountered and Solutions 90

8.2 System Strength 91

8.3 System Limitations 92

8.4 Future Enhancement 93

8.5 Knowledge and Experience Gained 93

8.6 Conclusions 94

References **96**

Appendix A- User Manual

List of Tables

Table 2.1: Different goals 8

Table 2.2: Comparison of different character sets used in Japanese. 9

Table 2.3: Summary of Reviewed Learning Packages 18

Table 2.4: Results from Surveying on Seniors Project 20

Table 2.5: Comparison between Visual Basic .NET and Visual Basic 6.0 24

Table 2.6: Microsoft Access 2003 general specifications 26

Table 2.7: Microsoft Server 2000 general specifications 28

Table 2.8: Major Operational and Design differences between Microsoft Access 2003 and Microsoft SQL Server 2000. 29

Table 2.9: Macromedia Flash MX and Director MX Comparison. 35

Table 3.1: Comparison of Command Life-Cycle Model 36

Table 4.1: Software Requirement to develop the system 50

Table 5.1 The Gane and Sarson notation 58

Table 5.2: Table of User 65

Table 5.3: Table of Administration 65

Table 5.4: Table of Lessons 65

Table 5.5: Table of HiraganaWriting 66

Table 5.6: Table of KatakanaWriting 66

Table 5.7: Table of KanjiWriting. 66

Table 5.8: Table of Exercises 66

Table 5.9: Table of Quizzes 66

Table 5.10: Table of Games 67

Table 5.11: Table of Scores for Quizzes 67

Table 5.12: Table of Scores for Games 67

Table 5.13: Table of Karaoke 67

Table 5.14: Table of Dictionary 67

Table 6.1: The Software tools for development 78

List of Figures

Figure 1.1: Project Schedule 4
Figure 2.1: Horizontal Text (write/read from left to right) 11
Figure 2.2: Vertical Text (write/read from right to left) 11
Figure 2.1: Kanji Trainer PenPen 13
Figure 2.2: Learn Japanese in Car 1-3 on cassettes 14
Figure 2.3: Declan's Japanese Flashcards 15
Figure 2.4: JquickTrans 16
Figure 2.5: Stackz 17
Figure 3.1: Incremental Evolutions through Rapid Prototyping 38
Figure 5.1: Architectural Design of Main Menu 52
Figure 5.2: Architectural Designs of Languages 53
Figure 5.3: Architectural Design of Maintenance Module 53
Figure 5.4: Architectural design of Lessons Module 54
Figure 5.5: Architectural design of Exercises Module 54
Figure 5.6: Architectural design of Quizzes Module 55
Figure 5.7: Architectural Design of Games Module 55
Figure 5.8: Architectural Design of Writings Module 56
Figure 5.9: Architectural Design of Dictionary Module 56
Figure 5.10: Architectural design of Karaoke Module 57
Figure 5.11: DFD of NihonGo- Level 0 59
Figure 5.12: DFD of User-Level 1 60
Figure 5.13: DFD of Lessons Module - Level1 60
Figure 5.14: DFD of Karaoke Module - Level1 61
Figure 5.15: DFD of Writing Module - Level1 61
Figure 5.16: DFD of Quizzes Module - Level1 62
Figure 5.17: DFD of Exercises Module - Level1 62
Figure 5.18: DFD of Dictionary Module - Level1 63
Figure 5.19: DFD of Games Module - Level1 63
Figure 5.20: DFD of Maintenance Module - Level1 64

Figure 5.21: Logical Database Design 68
Figure 5.22: Login Form for User 69
Figure 5.23: Home Form for User 70
Figure 5.24: Writings Module 71
Figure 5.25: Lessons Module 72
Figure 5.26: Exercises Module 73
Figure 5.27: Quizzes Module 74
Figure 5.28: Games Module 75
Figure 5.29: Karaoke Module 76
Figure 5.30: Writings Module (Malay version) 77
Figure 6.1: The Variable Declaration 80
Figure 6.2: The Code of Database Connection 81
Figure 6.3: The Code of SQL Statements 81
Figure 6.4: The Code of Exception Handling 81

List of Abbreviations

ADO	ActiveX Data Objects
CAI	Computer Aided Instruction
CAL	Computer Aided Learning
CD	Compact Disc
CD-ROM	Compact Disc Read-Only-Memory
DFD	Data Flow Diagram
DVD	Digital Video Disc
GIF	Graphic Interchange Formats
GUI	Graphical User Interface
ICT	Information and Communication Technology
IDE	Integrated Development Environment
MB	Mega Bytes
MCQ	Multiple Choice Questions
MPEG2	Moving Picture Experts Group 2
NihonGo	Japanese Learning Package
ODBC	Open Database Connectivity
OLE	Object Linking and Embedding
OOP	Object-Oriented Programming
RDBMS	Relational Database Management System
SQL	Structured Query Language
UI	User Interface
VCD	Video Compact Disc
XML	Extensible Markup Language

Chapter 1: Introduction

1.1 Project Initiation

Nowadays, there are a lot of Japanese teaching methods in the market for example one of the most popular ways of learning Japanese is through the Japanese self learning book. However there is a lot of limitations in this conventional learning method such as no pronunciation, no animation, and lack of color pictures due to the printing is expensive, thus it is why we need to develop this NihonGo system.

NihonGo is a Japanese learning package that teaches learners how to read and write Japanese language in hiragana, katakana and kanji. Every Japanese word comes with the English translation so that the learners can learn it easily. This learning package also teaches the pronunciation of each Japanese word as this is a method of teaching to overcome the weaknesses of learning Japanese language based on books alone.

1.2 Project Objective

The main objective of this project is to develop a computer-aided learning package for Japanese language. It would make the learning process as easy as ABC to the learners by providing the following functions:

a) **Writings:** Allows learner to learn and write Japanese writings in Hiragana, Katakana and Kanji.

b) **Lessons:** Learners can choose to learn the Japanese language step by step or according to lessons.

c) **Exercises:** Acts as tutorials for learners to enhance their understanding.

d) **Quizzes:** To test the learners' level of understanding and knowledge gained in

Japanese language and marks will be given and recorded in the top 10 table.

e) **Games:** To make the learning process interesting.

f) **Karaoke:** Provides Japanese songs' lyrics to help learners pronounce the Japanese words correctly and to make learning interesting. This helps to promote the users' interest in learning Japanese language and to memorize the Japanese language.

g) **Dictionary:** Learners can check for the vocabulary of the words they do not understand in the database.

h) **Maintenance:** Allows authorized administrator to update or modify functionality in NihonGo.

These functions would help to:

- Teach correct Japanese pronunciation.
- Enable the learners to speak in Japanese language.
- Teach the correct way of writing the Japanese characters.
- Provide interesting environment to learners.
- Test the learners' understanding of Japanese language.
- Enable learners to memorize the Japanese vocabulary and pronounce the Japanese words correctly through "Karaoke" singing.

1.3 Project Scopes

NihonGo is a Japanese learning package that targets for learners of age 12 years old and above and who is understand either English or Malay. This system is most suitable for beginners just starting to learn the Japanese language.

This is a bilingual system which is available in English and Malay. The contents are only limited to the beginner level which allows learners to read, write Hiragana, Katakana and Kanji and speak Japanese language and able to communicate in Japanese such as covers enough daily used words for daily used communication.

1.4 Project Outcome

This project is expected to develop a computer-aided learning package that would achieve the following:

- Provides simple, attractive and consistent background layout to attract the learners' attraction.
- Provides interesting, animated games and graphics to enhance the learning process.
- Text recitation is clear and understandable by learners.
- Allows adding of Japanese words into the database.
- Calculating of scores for the games, quizzes are accurate.

1.5 Project Schedule

In carrying out this project from June 2004 to end of January 2005, a precise project schedule is planned to manage the time and tasks that must be accomplished in the development phase. Figure 1.1 shows the tasks breakdown for the whole project within the estimated time required to complete each task.

ID	Task Name	Start	Finish	Duration
1	Project Initiation	6/28/2004	7/13/2004	12d
2	Literature Review	6/30/2004	7/30/2004	23d
3	Requirements Analysis	7/19/2004	8/13/2004	20d
4	System Design	7/28/2004	9/9/2004	32d
5	Module Development	8/20/2004	12/24/2004	91d
6	System Testing	10/22/2004	1/14/2005	61d
7	Maintenance	9/21/2004	1/28/2005	94d
8	Documentation	6/30/2004	1/31/2005	154d

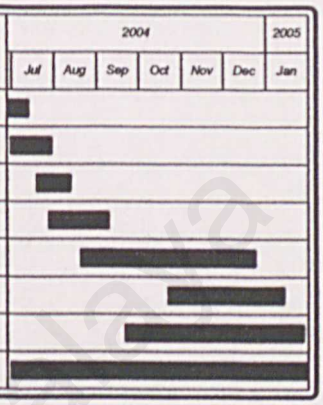

Figure 1.1: Project Schedule

Project Initiation

- Project understanding.
- Reviewed previous thesis reports done by seniors for reference.

Literature Review

- Analysis study on existing system development tools.
- Compare and select appropriate programming language, database, platform and multimedia tools for project.

Requirements Analysis

- Study and analyze the project features and review the suitable methodology.
- Understand how the system helps to solve problem.

System Design

- Translate requirements analysis into a system representation like architectural design, functional design, user interface design and database design.

Module Development

- System development and project implementation.
- Testing on completed sub modules.

System Testing

- Integrate sub modules into a full system.
- Main interface will integrate the entire sub modules.
- Test the project and fix the bugs found. Modification and changing done on any bugs found.

Maintenance

- Security and main database are required to complete the project.
- Post implementation reviews to determine the strengths and limitations of the system.

Documentation

- Information gathering conducted for each process and features.
- Complete package will be written in Compact Disc Read-Only-Memory (CD-ROM).

2.1 Introduction to Literature Review

Literature review surveys scholarly articles, books and other sources (e.g. dissertations, conference proceedings) relevant to a particular issue, area of research, or theory, providing a description, summary and critical evaluation of each work. The purpose is to offer an overview of significant literature published on the topic of project. In this chapter, literature review is conducted to gain a clear perceptive in developing this project. Surveys on existing available systems are also done to discover the weaknesses and strengths of the systems. Extensive researches on the available programming technologies, databases, multimedia tools and editing tools are done as to select the best way to fulfill system requirements.

2.2 Computer Aided Learning (CAL)

The applications of computers are growing rapidly, at the same time the expectations of this technology also growing at the same pace. One of the newest branches of this computer application is CAL.

CAL's history began in the early of 60's when the third generation of digital computers was being introduced. These systems were cheaper and more reliable than the previous models. Thus, it becomes the typical facilities in universities and research centers.

During the 70's the CAL systems were developed but the development rate still low due to some technical and economical limitations especially concerning hardware still expensive , massive and mainly without adequate sound and graphic facilities.

During the 80's and 90's the problems of CAL were eliminated by the digital evolution. Digital technology development upgraded hardware facilities and performances and with lower prices.

Nowadays, the current CAL system has the following features:

(a) Easy to access: Nowadays, every body cab easily get an educational package normally in a CD or uses educational websites.

(b) Quality: Sounds and graphic are really high quality.

(c) Storage: A normal CD can store up to 650 million letters (characters).

(d) System's variety and complexity.

CAL applications are not only necessary but is essential in the future seems the learning evolution, researchers believe that some factors are forcing to the evolution of current learning system: (Oblinger & Rush, 1997)

(a) Volume of knowledge – knowledge doubles every seven years.

(b) Technological competence – Adequate use of technological tools become a basic skills for employees.

(c) Re-skilling – a direct result from the two factors above.

(d) Group-working – or normally known as team-work.

(e) Home-based business.

The evolution of learning systems is also an adjustment to the working environment; Table 2.1 shows some differences between traditional education and working requirements:

Table 2.1: Different goals

Traditional Education	**Working Requirements**
Facts.	Problem solving.
Individual Effort.	Team skills.
Passing a test.	Learning how to learn.
Achieving grade.	Continues improvement.
Individual courses.	Interdisciplinary knowledge.
Receiving information.	Interacting and processing information.
Technology separate from learning.	Technology integral to learning.

The coverage areas of CAL's system in the future:

(a) Increasing system's performance – Includes higher speeds of processing and stronger software.

(b) More distributed systems – Tele conference and non-centralized classrooms will become a norm in the future.

(c) Simulation and virtual reality – Creation of 3 dimensional near-to-real environments for simulator systems such as the available systems of flight or sailing traing systems now.

(d) Special purpose CAL's system for the disabled – A multimedia workstation will can keep on repeating the words, and reduced the teachers' load sharply.

(e) Intelligent CAL system – Direct results of artificial intelligence methods.

2.3 Japanese

2.3.1 About Japanese Writing

There are three difference character sets used in Japanese there are katakana, hiragana and kanji. (Not include Ro-manji, Japanese written in Roman letters). Table 2.2 shows comparison between different characters sets used in Japanese.

Table 2.2: Comparison of different character sets used in Japanese.

Hiragana (ひらがな)	**Katakana** (カタカナ)	**Kanji** (漢字)	**Ro-manji**	**English**
とうきょう	トーキョー	東京	Tokyo	Tokyo

The **hiragana** and **katakana** (collectively called "kana"); each character represents the unique sound of a single syllable in the Japanese language. Hiragana and katakana both have about 46 characters that represent the different sounds possible in the language. Each sound is represented by a unique hiragana and katakana character. Although Hiragana and Katakana have the same pronunciation but they're used in different contexts. (Sterzinger, 1998)

2.3.2 Hiragana (ひらがな)

Hiragana is similar to the Katakana, also contains 46 basic symbols. All sounds in the Japanese language can be expressed with only hiragana. Each Hiragana symbol was simplified from a Chinese character that was being used to represent a sound but it much smooth, full of loops and curves. Hiragana was simplified from the radical, or sub-part of a Kanji. For example, the hiragana for "mi" (pronounced "me") み is based on the Kanji 美 meaning "beauty". And another example is the hiragana for "a"

(pronounced "ah") あ is based on the Kanji 安 meaning "peaceful". The function of the Hiragana is used for particles and copulas, and all inflectional endings and verb. (Takase, 2004)

2.3.3 Katakana (カタカナ)

Katakana contains 46 basic symbols. Each symbol was derived from a Chinese character in the same way as each Hiragana symbol, except that the Hiragana was simplified from entire characters. Katakana is used to write foreign words brought into Japanese, when the Japanese language lacked a native word to express a foreign idea. It might come from Chinese, Portuguese, Dutch and other languages. Katakana can be used to express any sound same with Hiragana. The first 5 characters correspond to 5 vowels common to many languages. (Sterzinger, 1998)

- A sounds like the "*a*" in the English word *father.*
- I sound like the "*i*" in the English word *machine.*
- U sounds like the "*u*" in the English word *truth.*
- E sounds like the "*e*" in the English word *prey.*
- O sounds like the "*o*" in the English word *most.*

2.3.4 Kanji (漢字)

The word Kanji (漢字) is the Japanese version of the Chinese word *hànzì*, which means "Han characters". Han refers to the Han Dynasty (206BC - 220AD) and is the name used by the Chinese for themselves. (Ager, 2004)

Kanji is the most complicated script in Japanese. First brought to Japan by Buddhist monks more than 1200 years ago. Thus the addition of Chinese characters to Japanese

increased in the number of concepts and methods of expression available to Japanese speakers, with the creation of many new terms and compounds.

Each kanji word might have different meaning, but not necessarily is read from different way. In fact, most of the characters have more than one way to pronoun. Such as, the character for *person,* 人 can be read as *jin, nin, hito, bito, ri,* and several other sounds. Readers need to know the differences pronunciation at the same character. (Sterzinger, 1998)

Since the Kanji is so complicate, there are small phonetic characters (known as Furigana) written beside of the Chinese characters when it write and they give the correct pronunciation of difficult or ambiguous readings. There can write in horizontal or vertical text, as shown in Figure 2.1 and Figure 2.2. E.g.: (Ager, 2004)

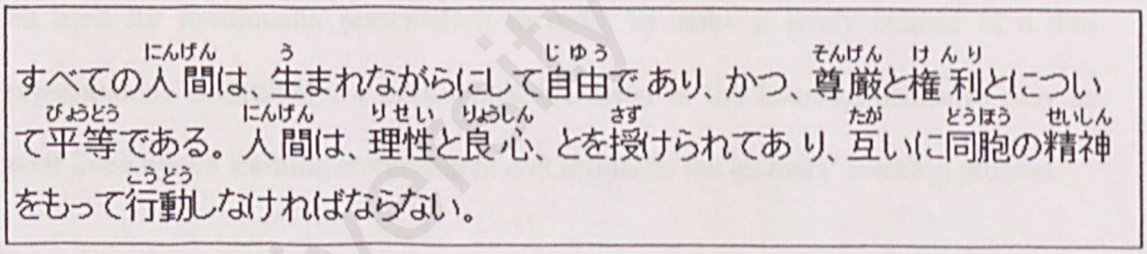

Figure 2.1: Horizontal Text (write/read from left to right)

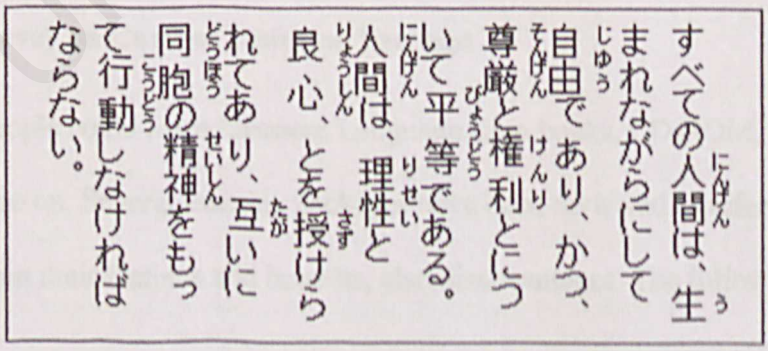

Figure 2.2: Vertical Text (write/read from right to left)

2.4 Multimedia

Multimedia elements can add exciting visual and auditory to attract viewers and to keep them excited with the interesting interfaces. The multimedia elements would include these interactive animated Graphic Interchange Formats (GIFs), Flash movies, Shockwave movies, Java Applets, and sound and video files.

With the advancement of technologies, learning using multimedia can be a solution for aiding learners. Using multimedia like CD-ROM, as a supporting tool for teaching Japanese language can be effective if implemented correctly and attractively. The advantage of a Japanese learning package in the type of CD is that a CD can store up to 700 Mb of data. This is equivalent to 500,000 pages of information in a book.

Nowadays, the 3D tools such as 3D Studio Max, Maya and Lightwave 3D have shown the full potential in game design and learning packages. Even these tools can be used for multimedia presentation in order to make it lively instead of a dull presentation. Undeniably, a multimedia inclusion in the learning packages may as well liven up the learning environment and influence the learners' learning process.

2.5 A Survey on Current Learning Package

Currently, people could learn Japanese Language from books, CD-ROM, VCD, website and so on. Several learning packages have been reviewed as references in this project to learn their features and benefits, also disadvantages. The following are the results:

I. Talk Now! - Learn Japanese (CD-ROM)

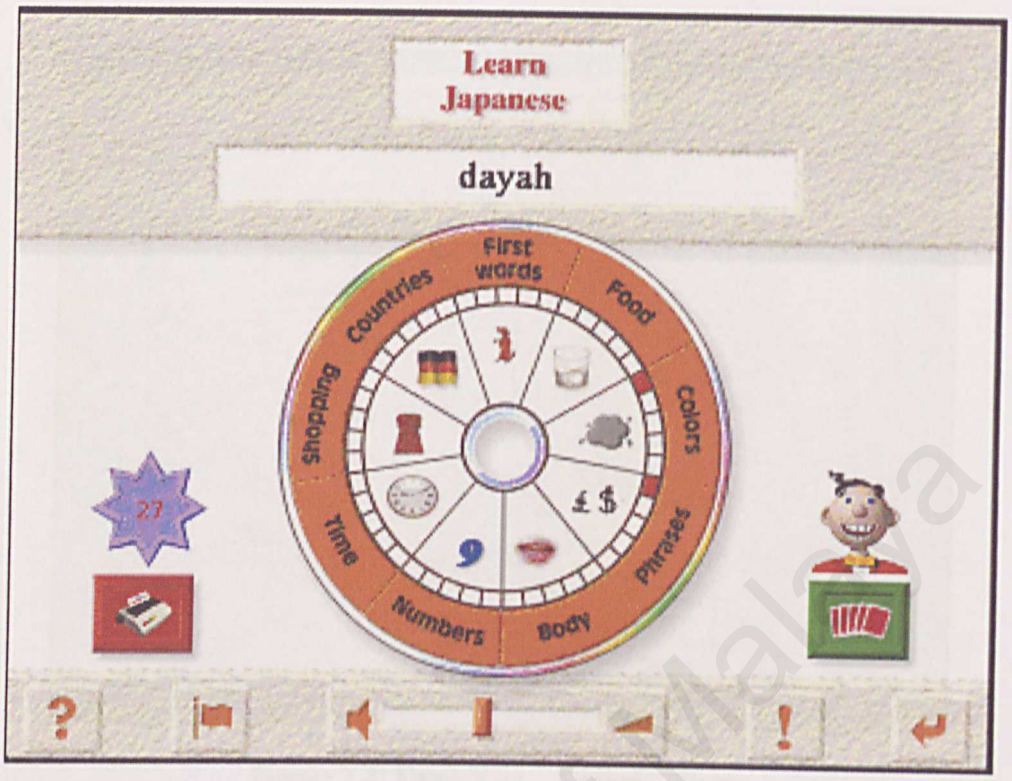

Figure 2.3: EuroTalk Interactive

Talk Now! is a beginning-level language course. It's a complete package comes with CD-ROM as distribution media. It represents only in English language. The disc includes nine topics: first words, food, colors, phrases, parts of the body, numbers, time, shopping and countries. This learning package license is supported for a user only. Each topic contains listening practice, an easy game, a difficult game, a printable dictionary and the opportunity to record your voice and hear how you sound in comparison to the two native speakers who tutor the user throughout the Talk Now! course. This learning package was promote in the website and available in Malaysia's market. (Euro Talk: Japanese, 2004)

II. Kana-Kun (CD-ROM)

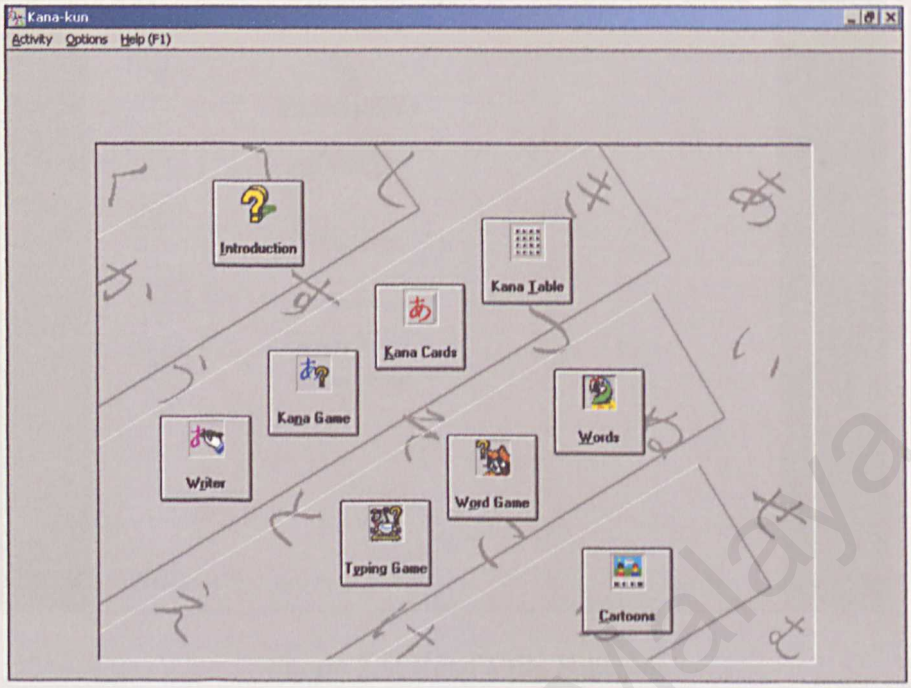

Figure 2.4: Kana-Kun Screen Shots

Kana-Kun is an advanced multimedia-learning tool for learning basic Japanese Language. The full version comes on CD-ROM and will teach all hiragana and katakana characters along with over 50 vocabulary words and a few simple phrases. Kana-Kun provides Kana tables, cards, writers and more functions to help user read and write hiragana and katakana properly. Also, this package comes with Word Game, Typing Game and more, which contain 200+ vocabulary words. This learning package was promote in the website and not available in Malaysia's market. (Kana-kun, 2004)

III. ReadWrite Hiragana

Figure 2.5: ReadWrite Hiragana

ReadWrite Hiragana teaches the Japanese Hiragana alphabet. Lessons include a structured introduction to each letter with examples of its pronunciation. Students can use the voice-recording feature to compare their pronunciation with a native speaker's. The Writing Lessons use animation to illustrate letter stroke orders. Students can practice writing and receive instant feedback on their efforts. Reading, Listening and Multi choice exercises are also included which are designed to build a simple vocabulary. This software is a part of Japanese Language Software package by Declan Software. The full packages offered alongside ReadWrite Hiragana are ReadWrite Kanji, ReadWrite Katakana, Japanese Flashcard and Japanese Dictionary. This learning package was promote in the website and available in Malaysia's market by registering the software product through online from. This program only runs in Windows 2000 and Windows XP. (Declan Software: Japanese, 2004).

IV. Basic Japanese Conversation (Book and Audio-CD)

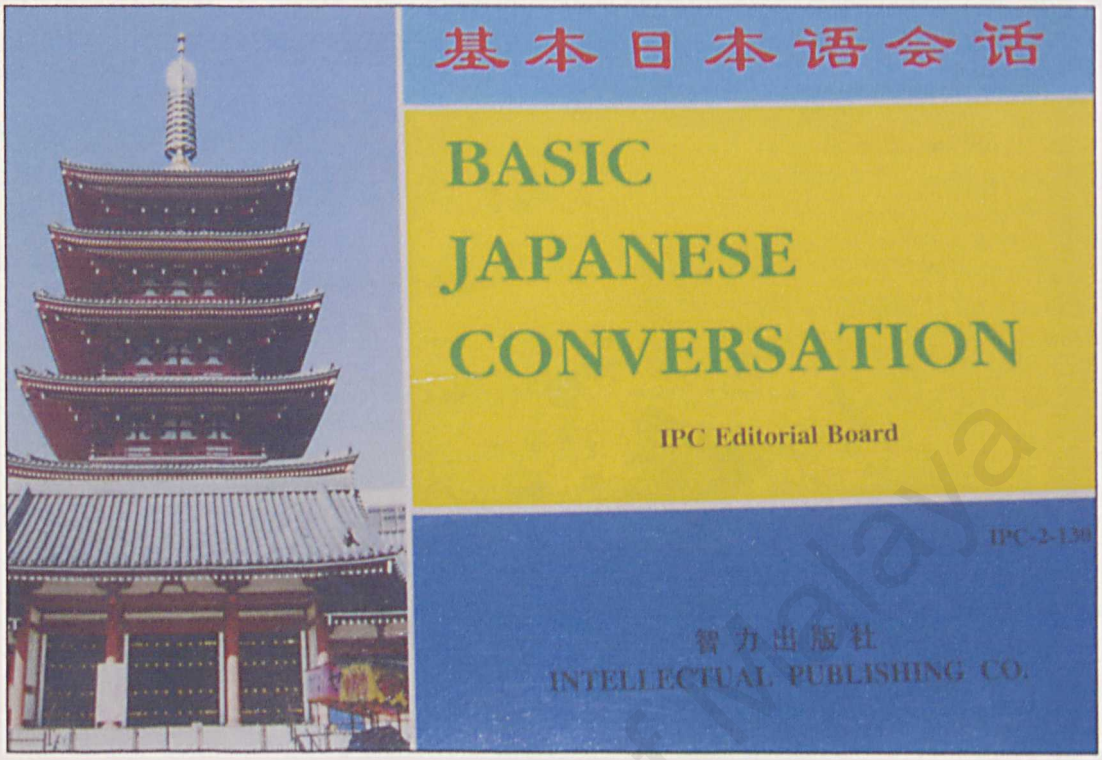

Figure 2.6: Basic Japanese Conversation

Basic Japanese Conversation is a book with Audio-CD for those who learn Japanese for the first time. This package contains 20 lessons. Each lesson consists of short daily dialogues, and contains a list of related words, which are very useful to form new sentences. This book also presents English, Simplified Chinese and Japanese Language. The Audio-CD contains every conversation in the lesson. This package is available in the market of Malaysia. (IPC Editorial Board., 2000).

V. Website

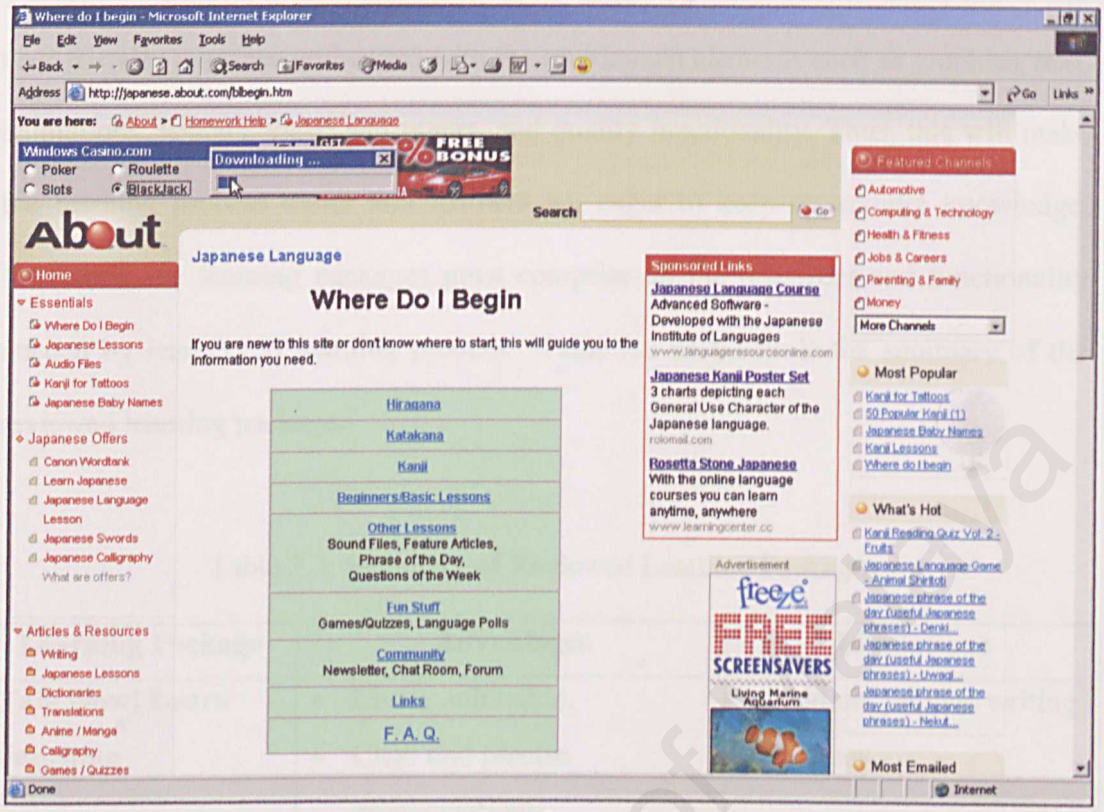

Figure 2.7: Website of Japanese Language Lessons

http://japanese.about.com/blbegin.htm is a website that offers Japanese Language lesson online. Also, this site provides great links to Japanese language resources. The guide for this site is Namiko Abe, whom graduated with a degree in English Literature from Kwansei Gakuin University, Japan. She has also studied teaching Japanese as a foreign language. (Japanese Language: Where Do I Begin, 2004)

2.6 Summary of Reviewed Learning Package

The most important characteristics for learning packages are to have an attractive and user-friendly interface embedded with the multimedia elements such as graphics, text, animations, sounds, video and others, and quality functionality. Thus, this will make the learning process easier and learners are eager to keep on acquire knowledge. Therefore, the learning packages must comprise all the resources and functionality needed by learners in learning process. Table 2.3 as below is the summary of the reviewed learning package.

Table 2.3: Summary of Reviewed Learning Packages

Learning Package	Advantages	Limitations
Talk Now! Learn Japanese	• Lively animation. • Clear and precise pronunciations. • Have functions to record user's voice. • Games and quizzes provided. • Interesting graphic user interface. • Have dictionary functions.	• Did not provide writing lessons.
Kana-Kun	• Use graphics to provide meanings. • Easy to use.	• Dull interface. • Less vocabularies and grammars lessons provided. • Only provide one type of game. • No dictionary function.

Learning Package	Advantages	Limitations
ReadWrite Hiragana	• Animation used to show how to write; stroke by stroke. • Examples on meanings of words are given.	• Only focusing on how to read and write. • Less grammars and particles lessons are provided. • No dictionary function. • Only focusing on read and write in Hiragana.
Basic Japanese Conversation	• Clearer and precise pronunciation. • Paper books are provided. • Books are in multiple languages: Japanese, Mandarin and English.	• No quiz and exercises provided. • Need books to learn words. • No dictionary to search.
Website	• Provide lessons for beginner in Japanese language. • Standardized layout.	• Non interactive. • Non multimedia. • Lessons are written in Ro-manji.

2.7 A Survey On Seniors Project

Besides surveying on existing available learning package for Japanese Language, research on Seniors Project has also been done. This research is to collect requirements on how to develop a good learning package. The results of this research are stated in Table 2.4 as below.

Table 2.4: Results from Surveying on Seniors Project

Title	Year	Description	Comment	Developing Tools
KBSR English Learning Package	1998/1999	An E-book, which consists of notes, quizzes and fun games for KBSR English.	• Simple instructions • Lack of animation	Visual Basic 5.0
ALPS – A Learning Package for SPM Paper 1	1999/2000	An E-book, which consists of notes, experiments, trail exam and exercises for SPM Physics.	• Attractive interface. • Provide attractive animation. • Provide online help.	Visual Basic 6.0, Microsoft Access 97, Flash.
Bilingual Learning Package for Opposite Meaning Words	1999/2000	An E-book, which consists of notes, quizzes and fun games for Opposite Meaning Words in English and Malay.	• Eye-catching Interface • Lack of animation. • Speeds of loading are slow.	Visual Basic 6.0, Microsoft Access 97
Chinese Wisdom	2000/2001	Computer aided learning package for Chinese Idioms.	• Attractive interface. • Lack of Test and Quizzes provided.	Visual Basic 6.0, Microsoft Access 97, Chinese Star 2.97
English Idioms	2000/2001	Computer aided learning (CAL) package for English Idioms.	• Interactive Interface • Poor quality of sound	Visual Basic 6.0, Macromedia Director 7, Flash 4

2.8 Consideration of Programming Language

2.8.1 Visual Basic .NET

At its core, Visual Basic .NET is an object-oriented programming language for writing applications that target Microsoft's .NET platform. Windows applications, Web applications, Web services, mobile applications, and more are possible with Visual Basic .NET. . NET is a software platform residing on top of the operating system that provides a structured hierarchical framework for building robust, secure, distributed applications in a variety of programming languages. It is said that 10 times real fast. Visual Studio .NET is an Integrated Development Environment (IDE) that natively supports Visual Basic .NET development. (Features Overview for Visual Basic .NET 2003, 2004)

Following are some of the basic features of the Visual Basic .NET language:

- Object-oriented.

  At its core, Visual Basic .NET is an object-oriented programming language that supports the .NET platform. It supports all major facets of any Object Oriented Programming (OOP) language-inheritance, encapsulation, and polymorphism.

- Structured.

  In addition to its OOP features, Visual Basic .NET supports typical structured programming constructs such as functions, procedures, conditional statements, loops, variables and arrays.

- Visual.

Visual Basic .NET natively supports the creation and usage of visual objects such as forms and Windows controls. It also supports the visual drag-and-drop ease of many modern-day programming environments. Instead of having to code many items, object properties can be set visually. Visual Basic .NET makes event-driven programming easy.

- New managed data providers.

Managed data providers deliver data controls for easy connections to OLEDB and ODBC data sources including Microsoft SQL Server™, Microsoft Access, Jet, DB2, Oracle, and more.

- Upgrade Wizard.

User is able to upgrade their code to receive all of the benefits of Visual Basic .NET 2003. The Visual Basic .NET Upgrade Wizard, available in Visual Basic .NET 2003 Standard Edition, and higher, upgrades up to 95 percent of existing Visual Basic code and forms to Visual Basic .NET with new support for Web classes and UserControls.

2.8.2 Visual Basic 6.0

Microsoft® Visual Basic® 6.0 is a language rapid application development environment that gives developers fast, easy, and intuitive tools to quickly develop Windows® applications. It provides a set of easy-to-use controls that programmers can use to develop interactive/ Window-based applications easily and quickly.

The "Visual" part in Visual Basic (VB) refers to the method used to create the graphical user interface (GUI). Rather than writing numerous lines of code to describe the appearance and location of interface elements, programmers simply added pre-built objects into place on screen.

VB programs are free standing programming environment. Windows programs can be complied and the programs can be exported to other Windows platform. VB environment is based on IDE where programming can be done in VB. The VB IDE has three states, which is Design, Run, and Debug.

Using Visual Basic, simple utilities or sophisticated applications can be developed. Data access features allowing user to create databases, front-end applications, and scalable server-side components for most popular database formats. Internet capabilities make it easy to provide access to documents and applications across the Internet or intranet from within user application, or to create Internet server applications. (Features by Edition for Visual Basic 6, 2004)

2.8.3 Comparison between Visual Basic .NET (.NET) and Visual Basic 6.0 (VB6).

Visual Basic .NET is an OOP language that supports the .NET platform. In addition to that, .NET supports the typical structure of OOP such as inheritance, polymorphism, multithreading, VB6 only supports some of OOP structure such classes and also supports the non-OOP structure. Background compilation help programmers to reduce syntax errors while coding. The .NET has built-in background compilation and VB6 does not support this feature. Another feature of .NET is built-in Package and Deployment Customization where developer is able to create installer .msi file, and customize deployment (e.g.: registry key value, product key). (Hoag, 2002)

Upgrade Wizard feature is only available in .NET as this feature allows developer to upgrade their VB applications to .NET version. Previous VB versions such as .NET 2002 and also VB6 are supported to integrate from their previous version to .NET.

Table 2.5 shows the comparison of .NET and VB6.

Table 2.5: Comparison between Visual Basic .NET and Visual Basic 6.0

Feature	Visual Basic .NET	Visual Basic 6.0
Structure	OOP	Some OOP, also support non-OOP
Background Compilation	Built-in	None
Package and Deployment Customization	Built-in	None
Upgrade Wizard	Built-in	None

2.9 Consideration of Database

2.9.1 Microsoft Access 2003

Access 2003 is a relational database management system (R-DBMS) which provides a powerful set of tools that are sophisticated enough for professional developers, yet easy to learn for new users. This R-DBMS create or use powerful database solutions that make organizing, accessing, and sharing information easier.

Access 2003 has automatic error checking, automatic property updating, and the ability to view object dependencies. Access 2003 also gives users expanded ability to import, export, and work with Extensible Markup Language (XML) data files. (Access 2003 Product Overview, 2003)

Below are list of some features with Access 2003:

- Incorporate a wide range of data sources.

Access 2003 supports a variety of data formats, including Extensible Markup Language (XML), OLE and Open Database Connectivity (ODBC).

- Link Business Systems.

Link tables from multiple databases at one time in Access 2003 forms, reports, and data access pages, which allow users to access any desired data. Link tables from other Microsoft Access databases, Microsoft Excel spreadsheets, ODBC data sources, SQL Server databases, and other data sources are also allowed.

- Deploy Access 2003 with triumph.

Access 2003 uses Access 2000 as the default file format for new databases, enabling users to deploy Access 2003 while maintaining capability with existing Microsoft Access users and solutions.

- Back up utilities.

Access 2003 also support back up critical database files to another location without leaving it, which helps prevent data loss by stopping the backup procedure if another user is using the database at the same time. As part of the backup operation, the Compact and Repair action is performed to make sure that the database is internally consistent.

The Microsoft Access 2003 general specifications are stated as Table 2.6.

Table 2.6: Microsoft Access 2003 general specifications

Object	Maximum sizes/ numbers
Microsoft Access file size (.mdb)	2GB
Number of objects in database	32,768
Number of characters in an object name	64
Number of characters in a password	14
Number of characters in a user name or group name	20
Number of concurrent user	255
Number of characters in a table name	64
Number of characters in a field name	64
Number of fields in a table	255
Number of open tables	2048
Table size	2GB
Number of characters in a text field	255
Number of characters in a memo field	65,535 / 1 GB
Size of an OLE Object Field	1 GB
Number of Indexes in a table	32

2.9.2 Microsoft SQL Server 2000

SQL Server 2000 is a fully enterprise-class database product, providing core support for Extensible Markup Language (XML) and Internet queries. It supports the rapid development of enterprise-class business applications. (SQL Server 2000 Product Overview, 2002)

Below are some features for Microsoft SQL Server 2000:

- Easy-to-use Business Intelligence Tools.

SQL Server 2000 provides critical, timely business information tailored to their specific information needs through rich data analysis and data mining capabilities.

- Self-tuning and management capabilities.

Revolutionary self-tuning and dynamic self-configuring features optimize database performance, while management tools automate standard activities. Graphical tools and wizards simplify setup, database design, and performance monitoring, allowing database administrators to focus on meeting strategic business needs.

- Data management applications and services.

SQL Server 2000 provides a powerful and comprehensive data management platform. It includes extensive management and development tools, a powerful extraction, transformation, and new capabilities such as Notification Services.

The Microsoft Server 2000 general specifications are listed in Table 2.7.

Table 2.7: Microsoft Server 2000 general specifications

Object	Maximum sizes/ numbers
Batch size	65,536 *packets
Bytes per short string column	8,000
Bytes per text, ntext, or image column	2GB
Bytes per Index	900
Bytes per foreign key	900
Bytes per primary key	900
Bytes per row	8,060
Connections per client	Maximum valued of configured connection
File groups per database	256
Files per database	32, 767
File size (data)	32 TB
Objects in a database	2,147,483,6474
Locks per connection	Max locks per server

2.9.3 Comparison between Microsoft Access 2003 and Microsoft Server 2000

SQL Server 2000 is an enterprise data management platform for organization that needs to adapt quickly in a fast-changing environment. Microsoft Access 2003 is a product from Microsoft that suites for desktop database management system. Access 2003 is the current version of this product whilst Microsoft SQL Server 2005 Express Beta.

Table 2.8 shows major differences in operational and designs between Microsoft Access 2003 and Microsoft SQL Server 2000.

Table 2.8: Major Operational and Design differences between Microsoft Access 2003 and Microsoft SQL Server 2000.

Feature	Microsoft Access 2003	Microsoft SQL Server 2000
Database size	2GB	32 TB
Objects in database	32,768	2,147,483,6474
Number of concurrent users	255	Maximum valued of configured connection
Failure recovery	Backup utilities – built in	Point in time recovery; failover cluster support
Security	Based on the use of workgroup information	Integrated with Windows 2000 and Windows NT security
Analysis Services (Data warehousing, data mining and OLAP)	None	Built-in
Support for SMP systems	None	Built-in

2.10 Consideration of Multimedia Tools

2.10.1 Macromedia Flash

Flash is an authoring tool that allows user to create anything from a simple animation to a complex interactive web application, such as an online store. Flash user can make their Flash applications media rich by adding pictures, sound, and video. Flash includes many features that make it powerful but easy to use, such as drag-and-drop user interface components, built-in behaviors that add ActionScript to their document, and special effects that they can add to objects.

When user author in Flash they work in a Flash document, a file that, when saved, has the file extension .fla. When users are ready to deploy their Flash content, they publish it, creating a file with the extension .swf. The Flash Player, described in the next section, runs the SWF file

Flash MX 2004 is the perfect tool for the web designer, interactive media professional, or subject matter expert developing multimedia content. Emphasis is on creation, import, and manipulation of many types of media (audio, video, bitmaps, vectors, text, and data). (Macromedia Flash MX 2004, 2004)

The new features in Flash MX 2004 provide greater productivity, enhanced rich media support, and streamlined publishing.

- Timeline Effects.

  User is able to apply Timeline effects to any objects on the Stage to quickly add transitions and animations such as fade-ins, fly-ins, blurs and spins.

- Behaviors.

With Behaviors, interactivity can be added to Flash content without writing a line of code. For example, user can use behaviors to include functionality that links to a website, loads sounds and graphics, controls playback of embedded videos, plays movie clips, and triggers data sources.

- Accessibility support in the authoring environment.

Accessibility support in the Flash authoring environment provides keyboard shortcuts for navigating and for using interface controls, letting users work with these interface elements without using the mouseUpload templates.

- Updates templates.

Flash includes updated templates for creating presentations, e-learning applications, advertisements, mobile device applications, and other commonly used types of Flash documents.

- Flash Player detection.

Developer can now publish SWF files with associated files that detect if a user has a specified Flash Player version. Developer also can configure their published files to direct users to alternate files if they don't have the specified Flash Player.

- Accessibility and components.

New accessibility features and a new generation of components offer tab ordering, tab focus management, and improved support for third-party screen readers and closed-caption programs.

- Globalization and Unicode.

New enhanced globalization and Unicode support allows Multilanguage authoring using any character set.

- String panel.

The new Strings panel makes it easier to publish Flash content in multiple languages. With the click of a few buttons, Flash creates external XML files for each specified language.

2.10.2 Macromedia Director 8.0

Macromedia Director MX is an authoring which allow user to build rich, interactive content and applications for CDs, DVDs, kiosks, and the Internet. Director handles the widest variety of media, letting user to build compelling, high-performance content—such as learning applications, corporate demonstrations, and games—that delivers measurable results.

User is able to customize their applications with Director's extensible plug-in architecture and ensure smooth playback, even when using long files or videos, with

Director's advanced memory management. Director offers powerful new features including a choice of scripting languages, DVD-video support, cross-platform publishing, and Flash integration. The new features in Director MX 2004 provide greater productivity, enhanced rich media support, and streamlined publishing. (Macromedia Director MX 2004, 2004)

- Powerful and flexible.

Director MX is extended in any direction with Director's powerful Xtras plug-in architecture. Xtras allow user extend the authoring environment and playback engine in nearly limitless ways. Also, incorporating new, powerful DVD functionality, which lets user embed, control, and play back DVD-video within multimedia projects.

- Integrated and streamlined.

Enhance overall performance with significantly improved Macromedia Flash playback performance within Director. Integrate function-rich, pre-packaged Macromedia Flash MX 2004 components, including user interface and media components.

- Productive and efficient.

Easily publish content across platforms and to different formats, including Macintosh and Windows projectors, in one simple step.

2.10.3 Comparison between Macromedia Flash MX and Director MX.

Macromedia Director has built-in advanced memory management, extensibility, and integration with high-quality media formats, such as DVD-Video, Apple QuickTime and Real Media. Intel Internet 3D graphics software included with Director. Advanced memory management enables optimal playback of fixed-media based multimedia files. The extensible Director architecture (Xtra extensions) provides the ability to add unique functionality for unlimited extensibility. Director MX 2004 supports over 40 different media formats, including QuickTime, Windows Media, and DVD, Flash content, Real Media and many others. Launch external applications from within Director for a seamless user experience. Accesses to file systems are available through Macromedia Shockwave Player by integrated security mechanisms. Director supports multiple input mechanisms, such as joysticks and the multi-button mouse, through the Xtra extension mechanism. Director offers a powerful, fully scriptable sound mixer, allowing low-level transformations of audio. Director offers a highly optimized rendering engine optimized for high-speed graphics and sound—both 2D and 3D.

Build rich Internet applications with a familiar forms-based development environment and powerful data binding with Macromedia Flash MX 2004. Macromedia Director MX 2004 enables the creation of accessible content that is available on Apple Macintosh and Microsoft Windows platforms, in browsers, or as a stand-alone executable, without the need for a screen reader. Macromedia Flash Player is the most widely distributed rich-media runtime, available on 98% of desktops. It offers a powerful scripting language for tracking user interactions, and seamless integration

with existing tracking systems. Macromedia Flash Player has been ported to many popular devices—including Pocket PC handhelds and leading set-tops—from vendors such as Liberate and Open TV.

Flash video is based on the H.263 video compression standard. Director MX 2004 supports DVD-Video, Windows Media, Real Media, Apple QuickTime, Flash video, as well as various other video formats. Through the Director MX 2004 runtime extension mechanism (Xtra plug-ins), various other formats (such as MPEG2) are also supported. Table 2.9 shows the summary of Macromedia Flash MX and Director MX comparison. (Macromedia Flash MX and Director Comparison, 2004)

Table 2.9: Macromedia Flash MX and Director MX Comparison.

Activity/Content	Macromedia Flash	Macromedia Director
Deploying to fixed media (CD/DVD, kiosk)		✓
Website user interfaces	✓	
Interactive real-time 3D		✓
Video: Long form	✓	✓
Internet-based rich-media applications	✓	
Accessible rich media	✓	✓
CD- or DVD-based applications		✓
Authoring & Runtime Extensibility		✓
Multiple media formats		✓
Launching external applications		✓
Access to file systems		✓
Support for devices in addition to webcam and microphone		✓
Precision sound control		✓
High-performance sprite rendering engine		✓
Online advertising	✓	
Content for devices	✓	

3.1 Introduction

This section will do the decision what methodology will going to use to develop the learning system. Many difference type of software life-cycle model has been reviewed from book and internet. There are no prefect methodology, every model have their strengths and weaknesses. Table 3.1 briefly shows the strengths and the weaknesses of some model. (Schach, 2003).

Table 3.1: Comparison of Command Life-Cycle Model

Lice-Cycle Model	Description	Strengths	Weaknesses
Code-and-fix model	·Implemented without requirements or specification. ·Simply throw code and rework until satisfy by client	·Fine for short program that require no maintenance.	·Totally unsatisfactory for products of any reasonable size.
Waterfall model	·Separate specification, development, validation and evolution to difference phases.	·Disciplined approach ·Document driven	·Delivered product may not meet client's needs
Rapid prototyping model	· Implement comes first. Once client satisfied the prototype, developers only can draw up the specification document.	·Make sure that the delivered product meet the client needs.	·Effort in building a prototype may be wasted
Iteration and Incrementation	·use multiple waterfall for increments	·Deliver functionality in distinct increments. ·Satisfy customer's need.	·The process is slow to deliver successive iterations of production code.

3.2 Incremental Evolution through Rapid Prototyping

The development model use to develop this project is Incremental Evolution through Rapid Prototyping (Figure 3.1). This model provides a way to use prototype in final system and be more responsive to customers quickly.

As shown in Figure 3.1, the prototyping incorporates the goals and opinions, requirements form the user from the beginning of the software evolution process to retirement, throughout the lifecycle. Prototypes develop very quickly, involving the user in an iterative build-execute-modify loop until the user is satisfied with the version of the prototype. This prototype is then used to build the final version of the software through the use of the architecture included in the prototype. The User requirements and specifications can be incrementally elicited and formalized through rapid prototyping. The final version is delivered relatively quickly.

Client might be change or add their requirements after delivered the final version. When the user's requirements do change, the final version became the base version of the product. Then the changes will make base on the base version until another final version is satisfied, then that will become the new version of the base version of the product. This incremental evolution process can proceed throughout the life of the system. (Dampier, 2004)

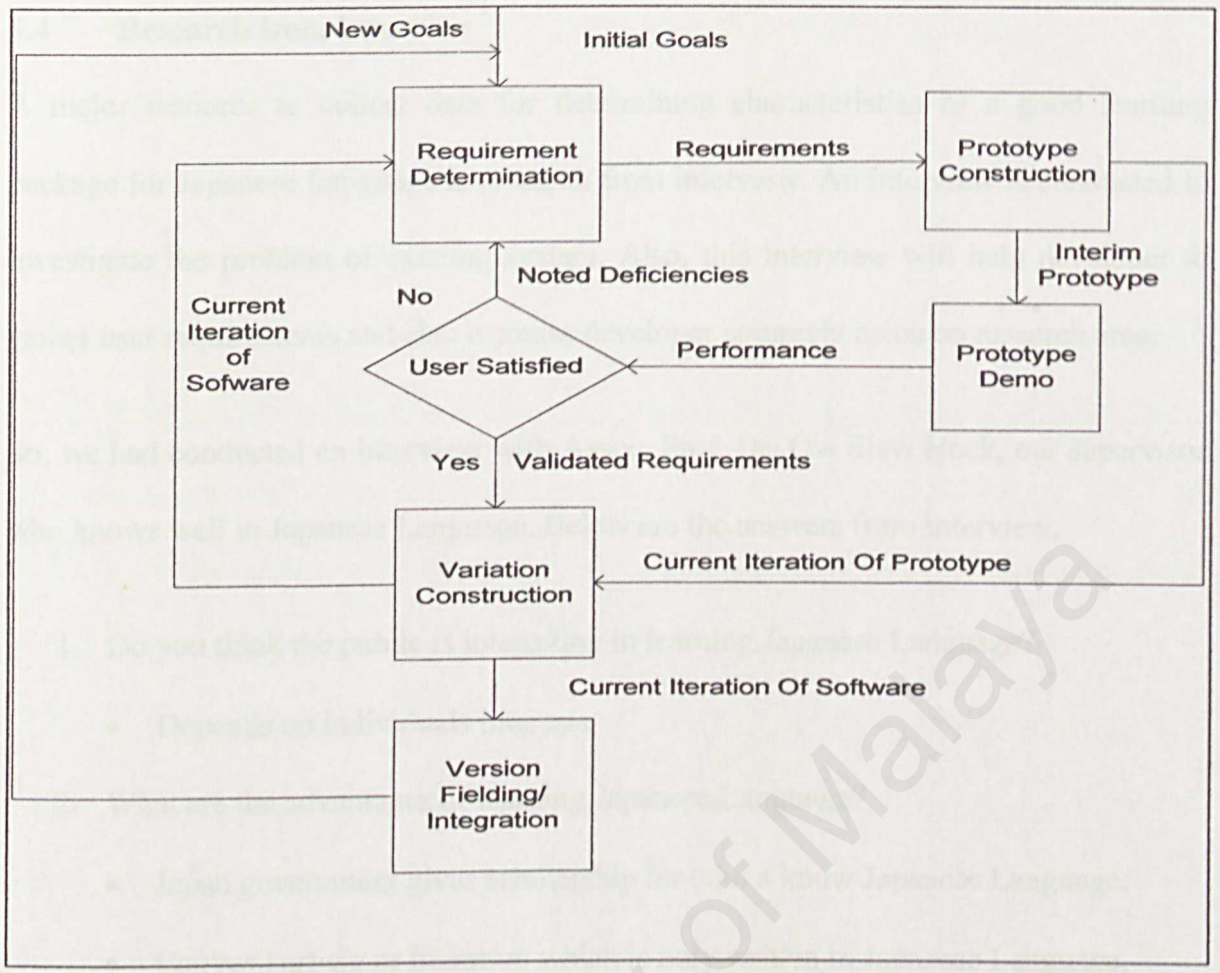

Figure 3.1: Incremental Evolution through Rapid Prototyping

3.3 Advantages in using the Evolution through Rapid Prototyping

- Aims to collect information on requirements and the adequacy of possible designs.
- Helps the designer to evaluate the design very early in the design cycle.
- It is good for addressing the problem of users not knowing or being unable to state their requirements.
- Provides the opportunity for continued evaluation and refinement of the design.
- Increases the chance of getting a well designed system acceptable to the users with the required functionality and ease of use.

3.4 Research from Interview

A major resource to collect data for determining characteristics of a good learning package for Japanese Language is research from interview. An interview is conducted to investigate the problem of existing system. Also, this interview will help developer to gather user requirements and also increase developer comprehension on research area.

So, we had conducted an interview with Assoc. Prof. Dr. Ow Siew Hock, our supervisor who knows well in Japanese Language. Below are the answers from interview:

1. Do you think the public is interesting in learning Japanese Language?
    - Depends on individuals interests.
2. What are the advantages of learning Japanese Language?
    - Japan government gives scholarship for who a know Japanese Language.
    - Can read article or literature which is only written in Japanese Language.
    - Easier to know the Japan culture.
    - Easier to get a job in Japan.
    - No problem if want to travel to Japan.
    - Can communicate with Japanese.
3. What are the problems that public faced in learning Japanese Language?
    - For non-Chinese educated, kanji is a big problem to learn.
    - If poor memory, will face problem to memorize the new language.
    - Less interest to learn new language.

4. In your opinion, what is the best method/effective way to learn Japanese Language?
    - Keep on practice, such as keep on write the new vocabulary, while writing, also repeat read it out as well.
    - Write each character in the small box, to train to write a good Japanese word.
5. What are the weaknesses in the current learning sources of Japanese Language learning tools?
    - Most of the source is book, which are no pronunciations provided and difficult to understand the grammar of the Japanese Language.
    - Furthermore, learners need waste a lot of time to keep on refers to dictionary.
6. What is lack in the learning package you have been use?
    - Tape with book as guideline.
        - Less explanation.
        - If not understand, need refer to the lecturer.
        - Only can use in lab.
7. Do we need to add Ro-manji at every lesson?
    - No, it will not help learner able to recognize the Japanese character.
8. For your opinion, what are the functionalities that can enhance the effectiveness of Nihon-go?
    - Accurate pronunciation.
    - Game, exercises and karaoke as well.
    - Dictionary.
    - Appropriate music and background.

9. What is your expectation to once learner complete learning Japanese from our learning tool?

- Leaner will able to write, read, listen and also able to conversation with people in simple Japanese Language.

3.5 Research from Internet

The Internet is a global network connecting millions of computers worldwide. More than 100 countries use this system to communicate data and information. Communications can several forms, such as e-mail, discussion groups, and information retrieved via the World Wide Web (also known as the Web). Much information on Japanese Language is collected through the Internet. On top of that, most of literature review is done through the Internet.

3.6 Research from Books

The traditional method to collect data is research from books. Many Japanese Language resources are gathered from books available in book stores. It helps to choose which lessons, exercises and sample are suitable and important to put in the system.

3.7 Research from the Market

This method was applied to investigate the similar computer-aided learning packages that are available in the local market. From this investigation, the strength and the limitation on the current software, tools had reviewed.

3.8 Research from Articles and Newspapers

There are many others method had used to get more information about Japanese Language and also the Japan culture as well. Some newspaper written in Japanese Language such as Nichi-Ma Press, Nangoku shinbun and Panora had get free distribute from the Jaya Jusco Supermarket. The pamphlet about the Japanese activity held in Malaysia subscribe by The Japan Foundation, Kuala Lumpur also collected. It all helps to more understand the Japan culture and society.

Review on senior's thesis report that has develops the learning tool system to give clearer picture of learning tool.

3.9 Why Use Visual Basic.NET

Visual Basic.NET is chosen because it can be used to produce "standard" Windows applications. .NET also supports typical structure of OOP such as inheritance, polymorphism and multithreading. The Windows applications that will be created still require a runtime module, but not the traditional VB runtime. Instead, any application that is created with Visual Basic .NET will be able to run only on host machines that already have the .NET redistributable installed on it. At press time, this redistributable package was just over 20 megabytes (MB) in size and was supported on Windows 98, 98 SE, Me, NT, 2000, and XP.

In addition to great Visual Basic .NET 2002 features are a powerful new forms designer, an in-place menu editor, and automatic control anchoring and docking, Visual Basic .NET 2003 delivers new productivity features for building more robust applications

easily and quickly. With an improved integrated development environment (IDE) and a significantly reduced startup time, Visual Basic .NET 2003 offers fast, automatic formatting of code as user's type, and also provides following features:

- Flexible, Simplified Data Access
- Background Compilation
- Full Object-Oriented Constructs
- Direct Access to the Platform
- Upgrade Wizard
- Package and Deployment Customization

3.10 Why Use Microsoft Access 2003

Microsoft Access 2003 is chosen because it can provide relational database power to manage information and also the easiest RDBMS in managing database. Access 2003 also lets user create and use powerful database management solutions that make organizing, accessing and sharing information easier. It also provides features for building tables, queries and forms that is simple and user-friendly. This can be adapted to this project needs and will be used as a Database Management System. Also, other features of Access 2003 are as following:

- **Incorporate a wide range of data sources.**

  Access 2003 supports a variety of data formats, including XML, OLE and ODBC.

- **View dependency information.**

  Quickly find tables, queries, forms, or reports that depend on a particular database object

- **Update properties automatically**

  Change a field property in a table and all the forms or reports that have controls bound to it can be updated automatically.

- **Use compatible file formats**

  Access 2003 uses Access 2000 as the default file format for new databases because Access version 2002 and Access 2000 can use and modify the same database.

3.11 Why Use Macromedia Flash.

Macromedia Flash is chosen of its ability generate .swf file format to deliver graphics, animation and sound. Almost 95% of users can view .swf contents without having to install a new plug-in.

Flash have improve design and layout, animation navigation and user interfaces. Also, features interactive applications with Behaviors element (reduce the need to script simple tasks such as media and navigation controls) and jump starts with drag and drop User Interface (UI) components (a series of UI building blocks such as buttons, menus, and

scrollbars). Flash also includes additional UI components as well as media components, enabling user to quickly and easily incorporate video into their projects.

Flash is able to create common timeline animations with just one step, reducing the need for excessive key framing. Effects are non-destructive so user can modify them repeatedly or undo them. Animate and control objects over time with an advanced timeline. Set up content efficiently with multiple layers, motion guides, and layer folders are done using timeline feature. Flash is based on key frame animation for stop-motion, shape, and property or motion tweening with easing control.

3.12 Why Use Adobe Photoshop

Adobe Photoshop is a program to create high quality image with significant new creative tools, workflow options and editing functions. It is integrated with Macromedia Flash file export to create .swf animations. In addition to this, the function is complete vector artwork and variable text.

Photoshop is integrated with scan function and this helps especially on editing scanned graphics. This provides user with high quality images and graphics with smallest possible size.

This powerful graphic editing program is available on wide array of platforms ranging from Windows to Mac. Another advantages in using Photoshop is the complete set of graphic editing tools, the most intuitive user interface provided and the largest number of reference books and online tutorial are available. (Adobe Photoshop CS, 2004).

Chapter 4: System Analysis

System analysis is the structured analysis of a process, with strong attention paid to the learners' needs, which determines the program's current operating condition and yields objective data to ensure the continuous improvement of the program. The analysis is made on the system architecture, the functional requirements and non-functional requirements. All of these analyses are very important for the following phase, system design.

4.1 Functional Requirement

Functional requirements are statements of system's services that specify an action that the target product must be able to perform. They are often expressed in terms of inputs and outputs. If given a specific input, the functional requirement stipulates what the output must be and how the system should behave in particular situations.

i. Maintenance Module (Administrator only).

Authorized developers are allowed to add more exercises and notes into this module.

ii. Writings Module.

The application of writing is to enable learners to learn and write Japanese character in Hiragana, Katakana and Kanji. The correct sequence order of writing Japanese characters is shown by animated graphics. The learners can learn writing in the system using mouse.

iii. Lessons Module.

Learners are allowed to choose the languages they prefer whether the translation is in English or Malay language. The system will have a text-to-speech function to teach them the correct way of pronunciations during the 20 lessons course in the syllabus.

iv. Exercises Module.

In exercises module, learners can choose an exercise to answer based on the lesson they study. There are 4 types of questions: Objective Questions, Multiple Choice Questions (MCQs), Fill in The Blanks with On-Screen Keyboard, and True or False Questions. Each exercise has 10 questions. As the system has a test bank that stores more than 10 questions, the exercises' questions are generated randomly. With 20 lessons, there are 20 exercises with answers provided. No marks are given in exercise module. .

v. Quizzes Module.

In quizzes module, learners can choose a quiz to answer based on the lesson they study. There are 4 types of questions: Objective Questions, MCQs, Fill in The Blanks with On-Screen Keyboard, and True or False Questions. Time allotments for Chapter 1 to 10 and Chapter 11 to 20 are 10 minutes and 15 minutes respectively. Each quiz has 10 questions. As the system has a test bank that stores more than 10 questions, the quizzes' questions are generated randomly. When the time is over, the quiz is automatically stopped from answering. Marks are given and the quiz result is saved into database before displayed in the table. Top 10 results are shown in the performance table.

vi. Games Module.

The simple and interesting games in learning package are good in relaxing learners' mind and evaluating the understanding of learners. In order to attract the learners, numerous graphics, animations and sound effect are needed to make the games jovial. Learners must understand the Japanese language before playing the games. There are 5 games in Games module.

vii. Karaoke Module.

To make it more interesting, a karaoke module is added to let learners learn to sing in Japanese language. Singing in Japanese language can help learners to speak Japanese fluently. Lyrics and the translations are part of the karaoke module. Eight songs are available to let learners choose for themselves. Thus, learners can enjoy and learn to sing the song at the same time.

viii. Dictionary Module.

In helping learners to fully understand Japanese language, a dictionary module is essential in letting the learners to know the meaning. At different level, learners may sometimes forget the meaning, thus they can search through the dictionary. The learners can search the meanings either by key-in ro-manji or hyperlinks of certain vocabulary.

4.2 Non-Functional Requirement.

Non-functional requirements presents a systematic and pragmatic approach to "building quality into" software systems. Software quality attributes such as accuracy, performance

and security must be exhibit from a system. Such non-functional requirements are difficult to address and modeled since it has a broad impact on the system. The constraints are execution constraints, development process constraints, standards constraints, etc.

i. User Friendly.

The system's interface should be friendly, attractive and easy to use. The system's functionality and purposes must be consistent within themselves. In order to attract the learners' attention, interesting animation and colorful graphics are vital in the system.

ii. Correctness.

The translation and information display in this learning package should be accurate and correct. Each translation from Japanese to English or Malay languages have to represent the exact meaning of the Japanese language. The way to approach this requirement is experience and professional people, like a translator, should check the translation.

iii. Extensibility.

The ability to accommodate increased functionality in standalone software is necessary if there is any update to it. Function such as embedded Japanese e-learning can be considered in future enhancement as the system is built in separate modules. So, there is modularity in our system to enable system extensibility. Thus, NihonGo must accommodate any requirements arise in future development.

iv. Security.

The system's accessibility must only be given to the authorized administrator for any update and maintenance. Thus, an administrator can correct the system if there are any faults after installation. Unauthorized person can disrupt the system only if there is no security.

v. Understandability.

The programs' code is written by using natural language such as English. It is to let other programmers understand the data flow and coding method used in developing this learning package. The inclusion of comments is used to help programmers understand the code and its flow.

4.3 Software Requirement

i. Developer Software Requirement.

Literature and research review were completed at the initial phase. The extensive research on the existing programming technologies, databases, multimedia tools and editing tools have brought the decision to the tools usage as shown in table 4.1

Table 4.1: Software Requirement to develop the system

Operating System	Microsoft Windows XP
Programming Language	Visual Basic .NET
Multimedia Tools	Macromedia Flash MX
Editing Tools	Adobe Photoshop CS
Database	Microsoft Access 2003

ii. User Software Requirement.

These are the software required by user to use this system:

- Windows 9x/2000/XP.`
- Microsoft Access.

4.4 Hardware Requirement

i. Developer Hardware Requirement.

After reviewing on most of the hardware requirement on Japanese learning package system, these are the minima requirement to develop a learning package system:

- Intel Pentium 233MHz or better.
- 128 MB RAM or more.
- 500 MB of available disk space.
- 16-bit color monitor (1024 x 768 resolution).
- 16 bit sound card.
- CD-ROM drive.
- Mouse, speaker, microphone.

ii. User Hardware Requirement.

For smooth performance of the system, users are advisable to fulfill the requirements as below:

- Intel Pentium 200 MHz or better.
- 128 MB RAM or more.
- 250 MB of available disk space.

5.1 Architectural Design

The architectural design is a pictorial representation of the subsystem framework control and communication. It is system module decomposition into several subsystems with each subsystem has the produced functionality. The architecture describes the interconnections among the modules.

Figure 5.1 shows the interconnections between the independent modules for the Japanese language learning process. The main system is divided into 7 modules with each of the modules being broken into detail sub-modules. User has the option to choose the language they want to display.

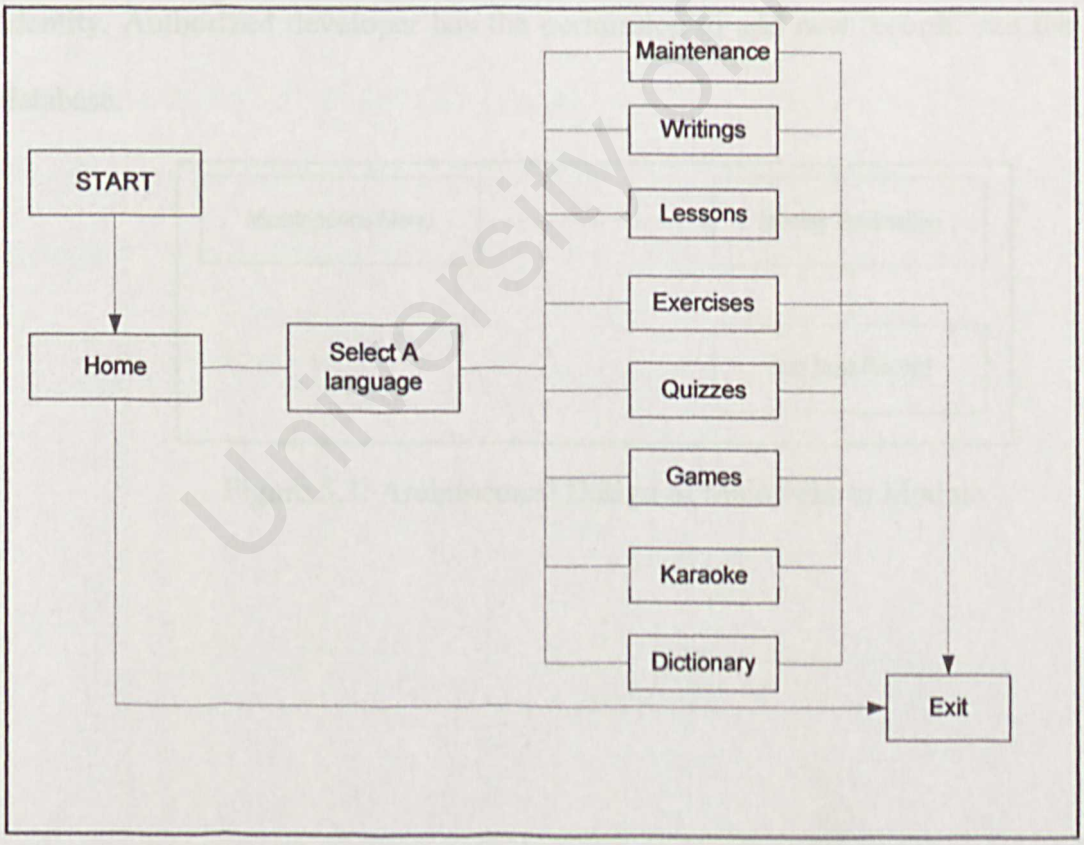

Figure 5.1: Architectural Design of Main Menu

Figure 5.2 shows a part of the system that allows a user to choose the language they prefer. The options are English, Malay languages. This part is available in every page in order to let the user changes the language flexibility.

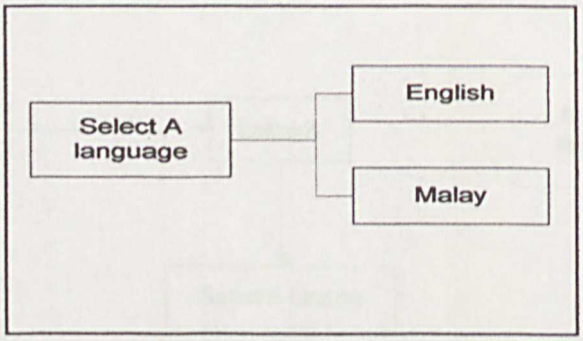

Figure 5.2: Architectural Designs of Languages

In figure 5.3, to avoid unauthorized access, a system needs to verify the developer's identity. Authorized developer has the permission to add new records into the system's database.

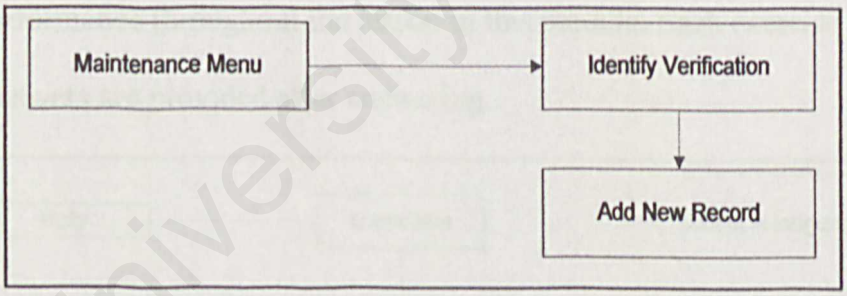

Figure 5.3: Architectural Design of Maintenance Module

Figure 5.4 is the most important part in the Japanese language learning process. A translation and text-to speech functionality for Japanese words are provided for the user's learning easiness. There are 20 lessons in this module.

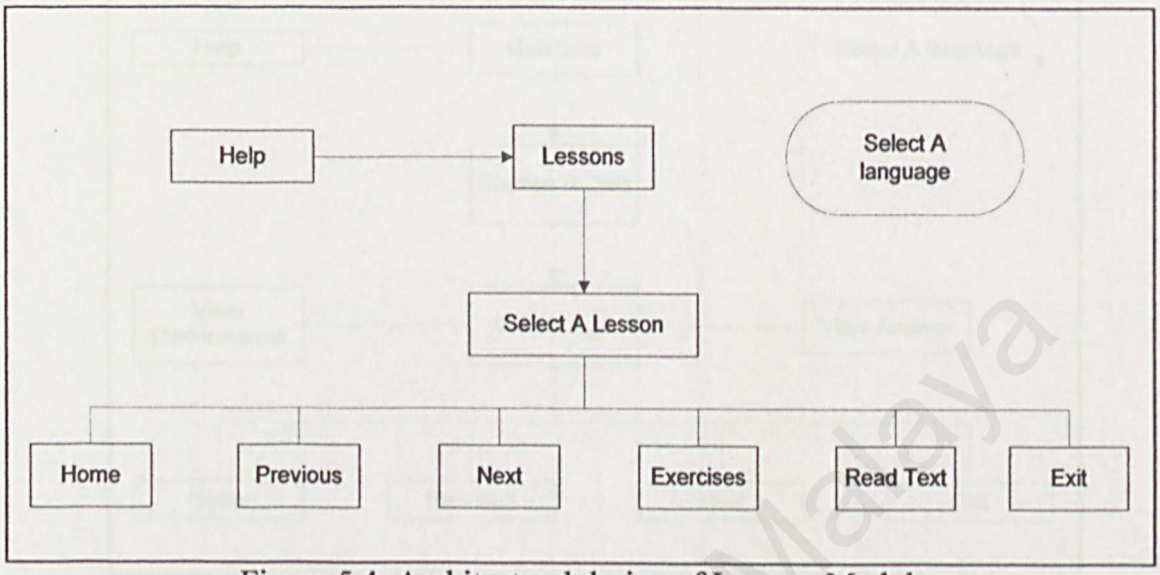

Figure 5.4: Architectural design of Lessons Module

Meanwhile, in figure 5.5, the exercises of each lesson are displayed here. The user can view their performance throughout the lesson in this module. Each exercise consists of 10 questions. Answers are provided after answering.

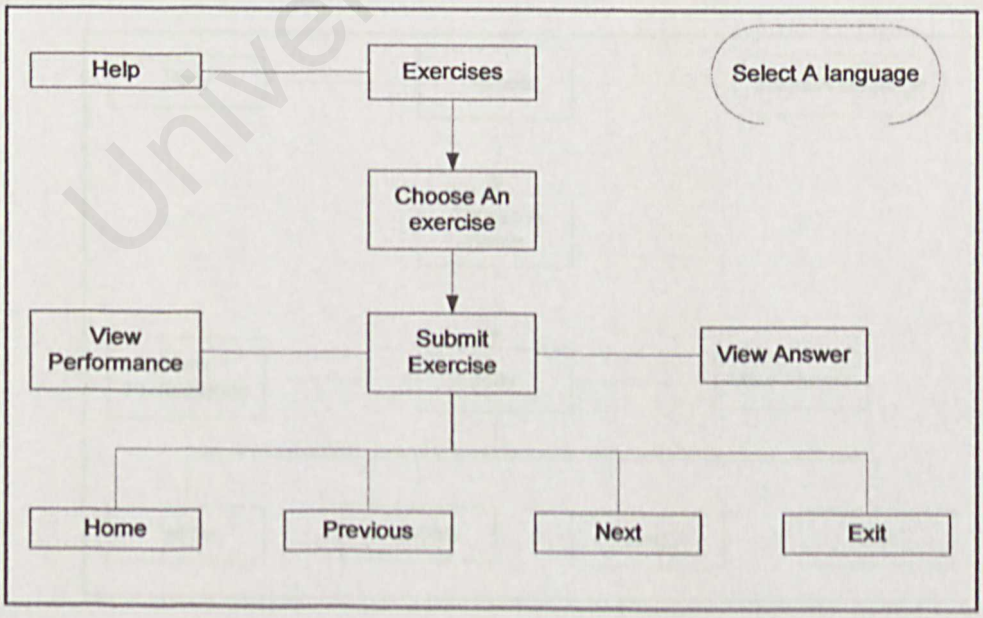

Figure 5.5: Architectural design of Exercises Module

Meanwhile, in figure 5.6, the quizzes of each lesson are displayed here. The user can view results throughout the lesson in this module. There are 10 questions for each quiz. Marks are given.

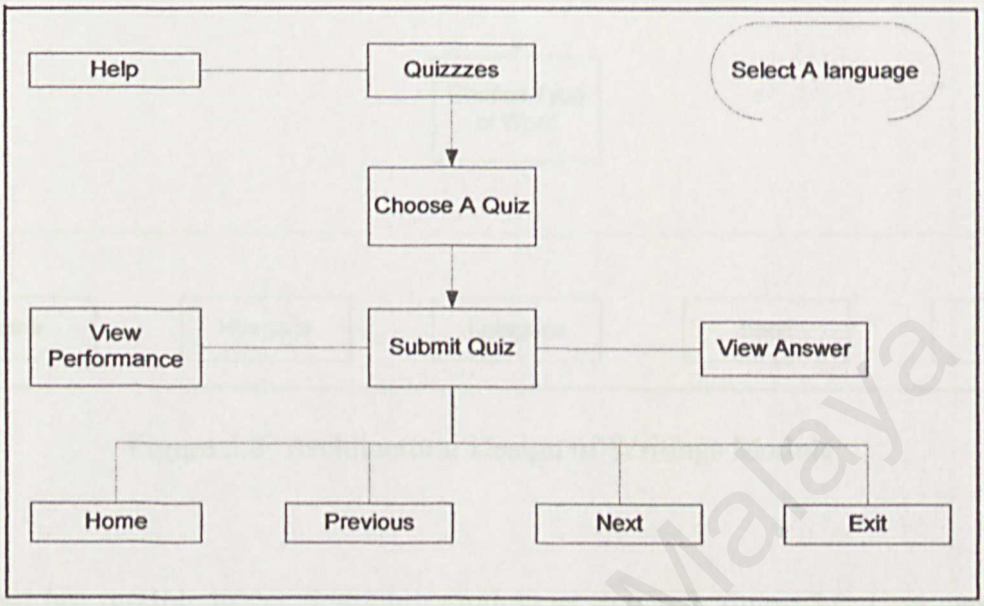

Figure 5.6: Architectural design of Quizzes Module

The game module is shown in figure 5.7. The user can choose a game that they are interested. After that, the user can view their performance in the performance part.

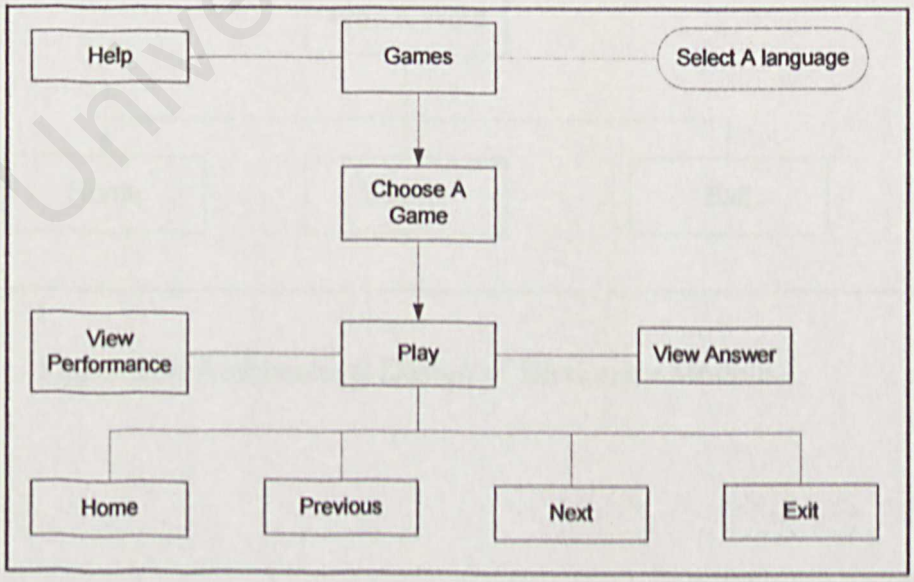

Figure 5.7: Architectural Design of Games Module

In figure 5.8, a writing module is provided for user to practice learning the correct way.

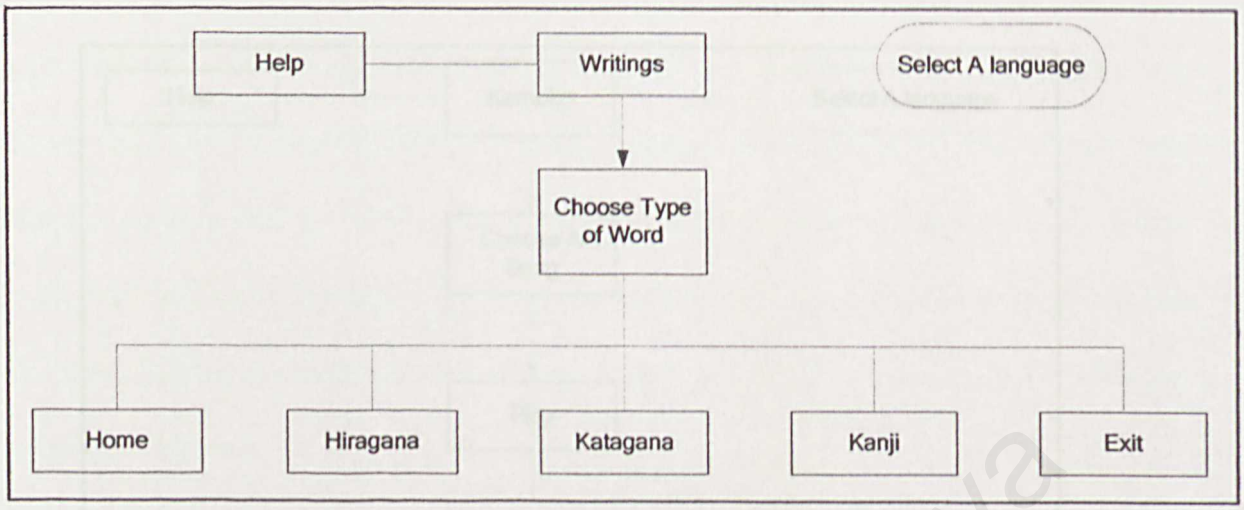

Figure 5.8: Architectural Design of Writings Module

The second last module is the dictionary module as shown in figure 5.9. User can search for the meanings for words they want. A translation of each word is shown in the system.

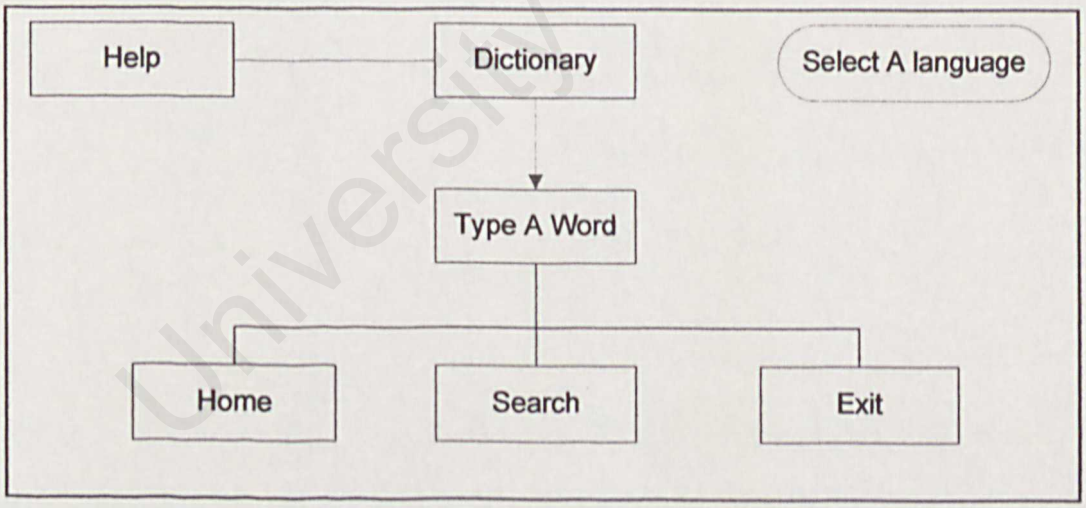

Figure 5.9: Architectural Design of Dictionary Module

The last module is the karaoke module as shown in figure 5.10. Users can select the songs they like. Lyrics will be provided so that users can sing along.

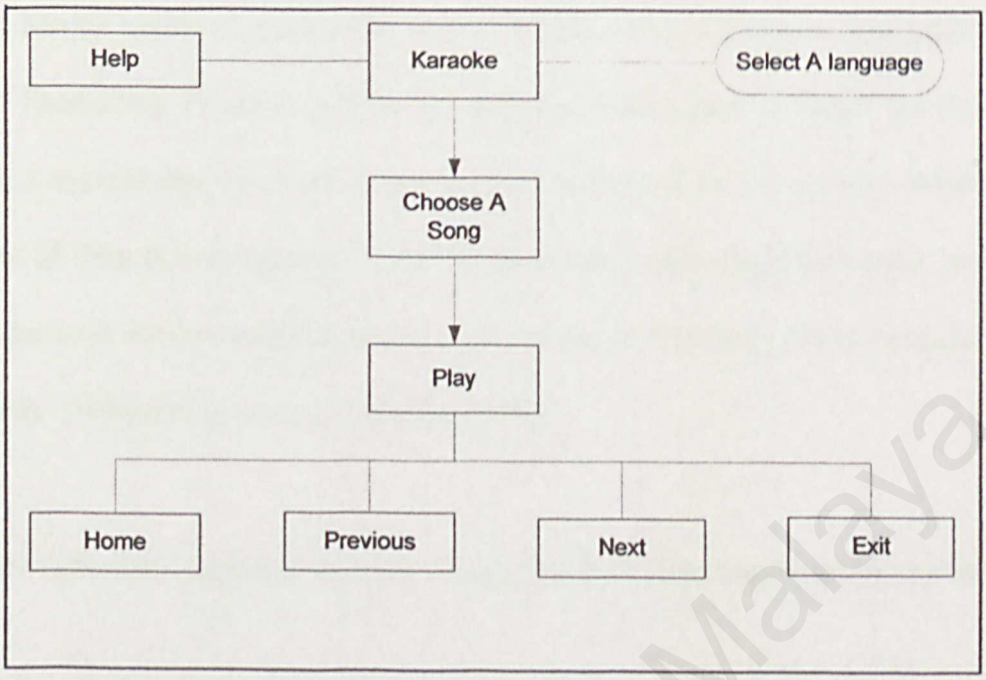

Figure 5.10: Architectural design of Karaoke Module

5.2 Process Model

Process Model involved graphically characterization for function or data processes for a system. Data Flow Diagram (DFD) is a process model used to depict the flow of data through a system and the work or processing performed by the system. Whitten shows the flow of data or information. It can be partitioned into single processes or functions. The Gane and Sarson notation had choose to use in following DFD because it's wide popularity. (Whitten, Bentley & Dittman, 2000).

There are only three symbols and one connection in the Gane and Sarson notation:

Table 5.1: The Gane and Sarson notation

Symbol/Connection	Details
Process	A rounded rectangle (the Gane and Sarson notation) is used to represent a process.
Agent	Square represent external agent.
Data	The open-ended boxes represent data stores.
information	The arrow represents data flows, or input and outputs to and from the process.

DFD of every module of this project are shown from Figure 11 to 20.

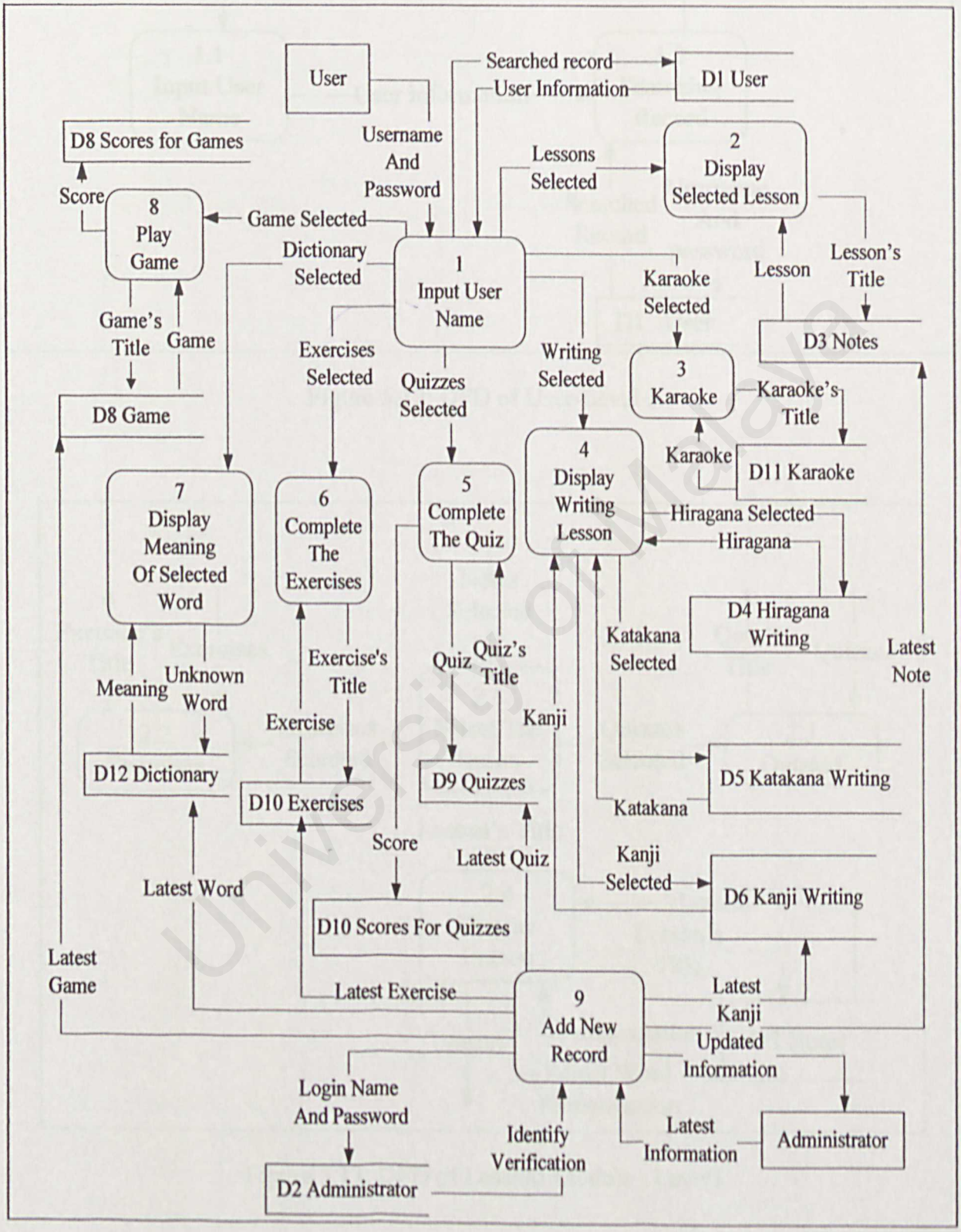

Figure 5.11: DFD of NihonGo- Level 0

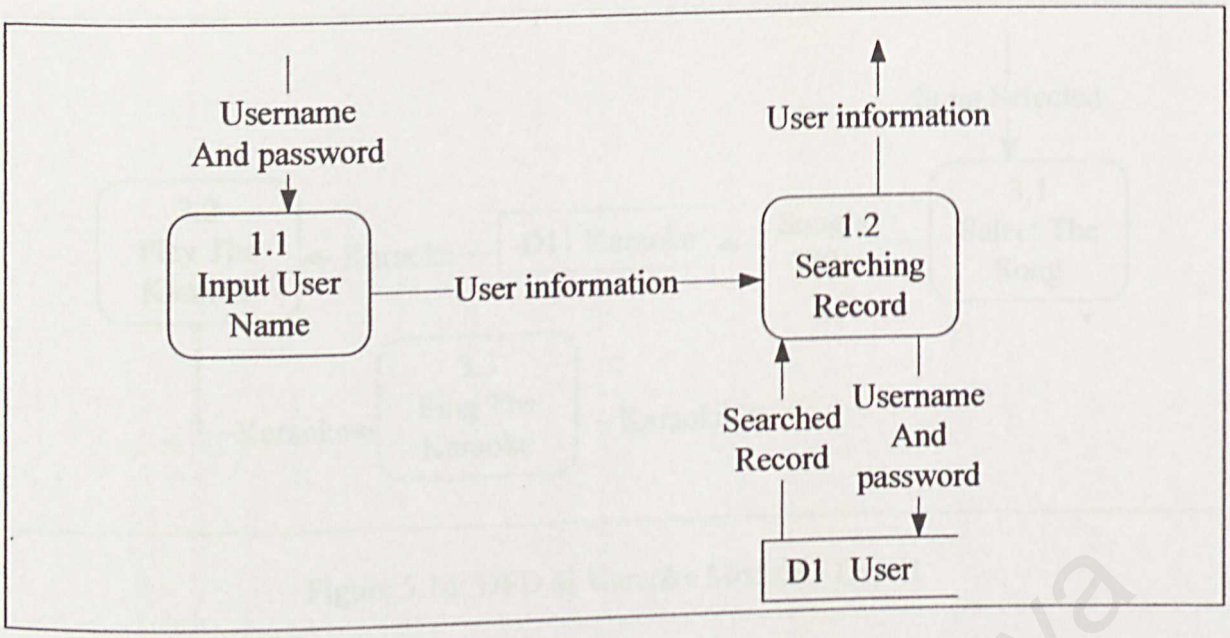

Figure 5.12: DFD of User-Level 1

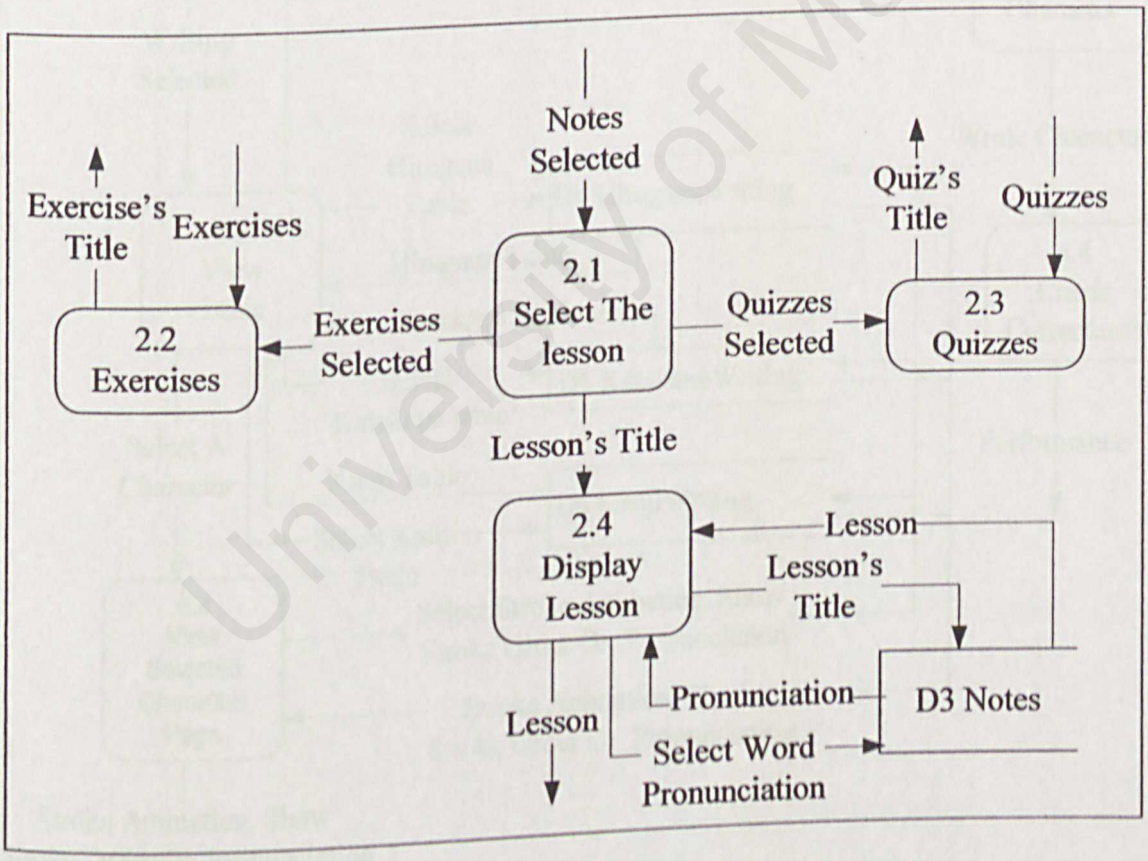

Figure 5.13: DFD of Lessons Module - Level1

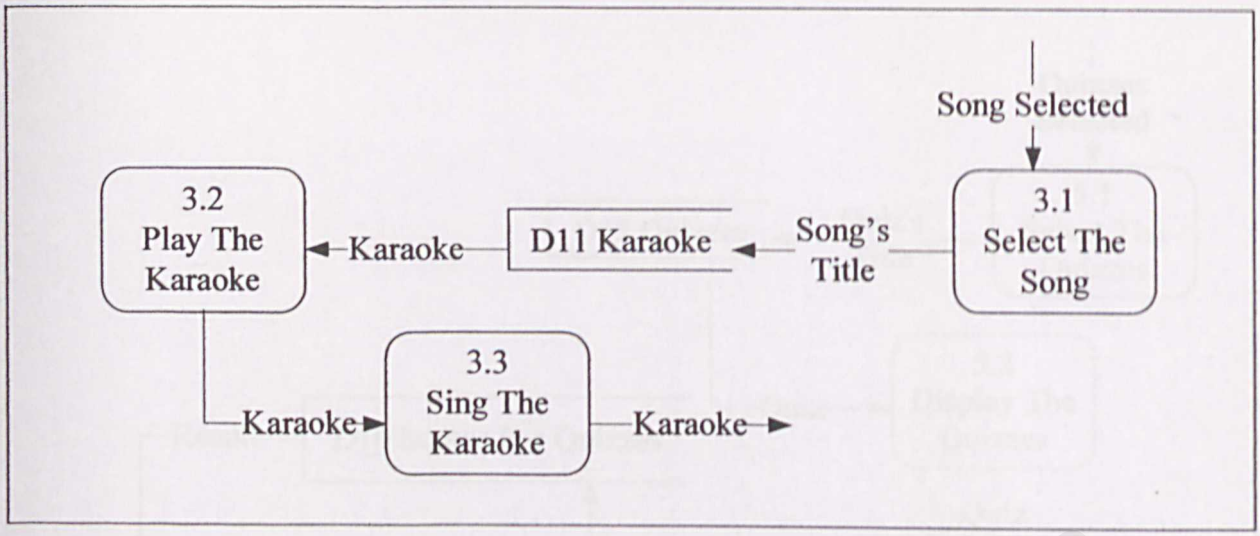

Figure 5.14: DFD of Karaoke Module - Level1

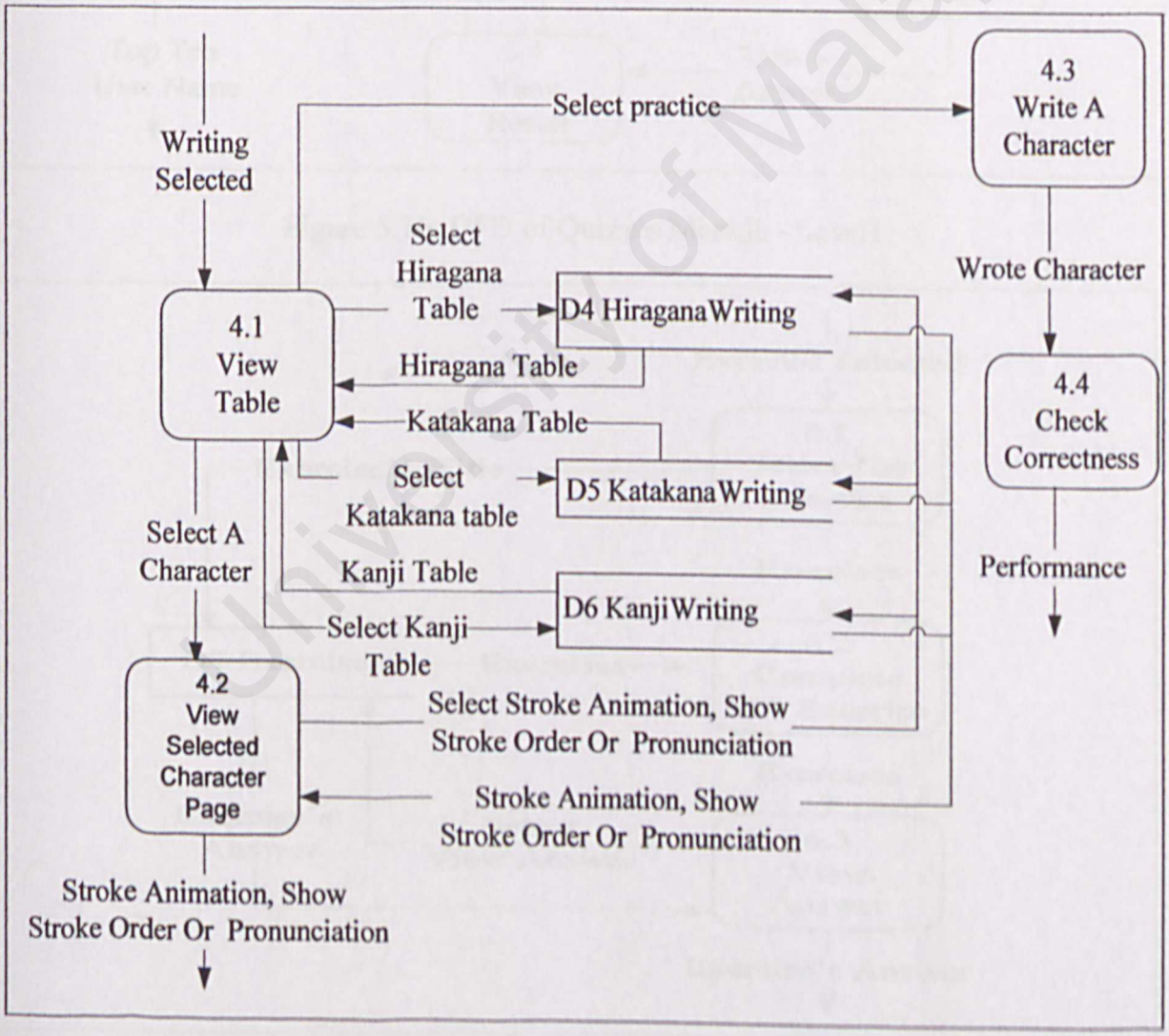

Figure 5.15: DFD of Writing Module - Level1

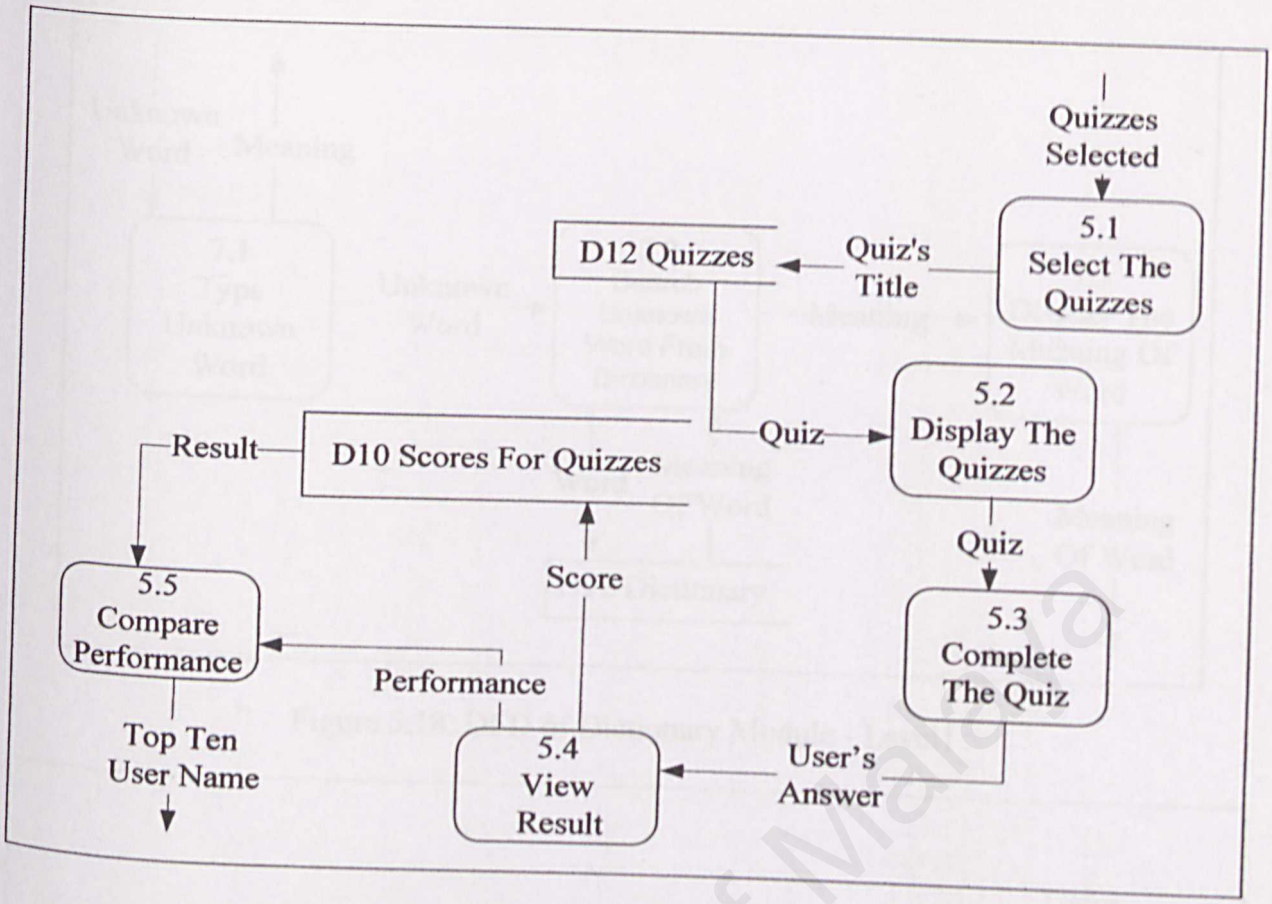

Figure 5.16: DFD of Quizzes Module - Level1

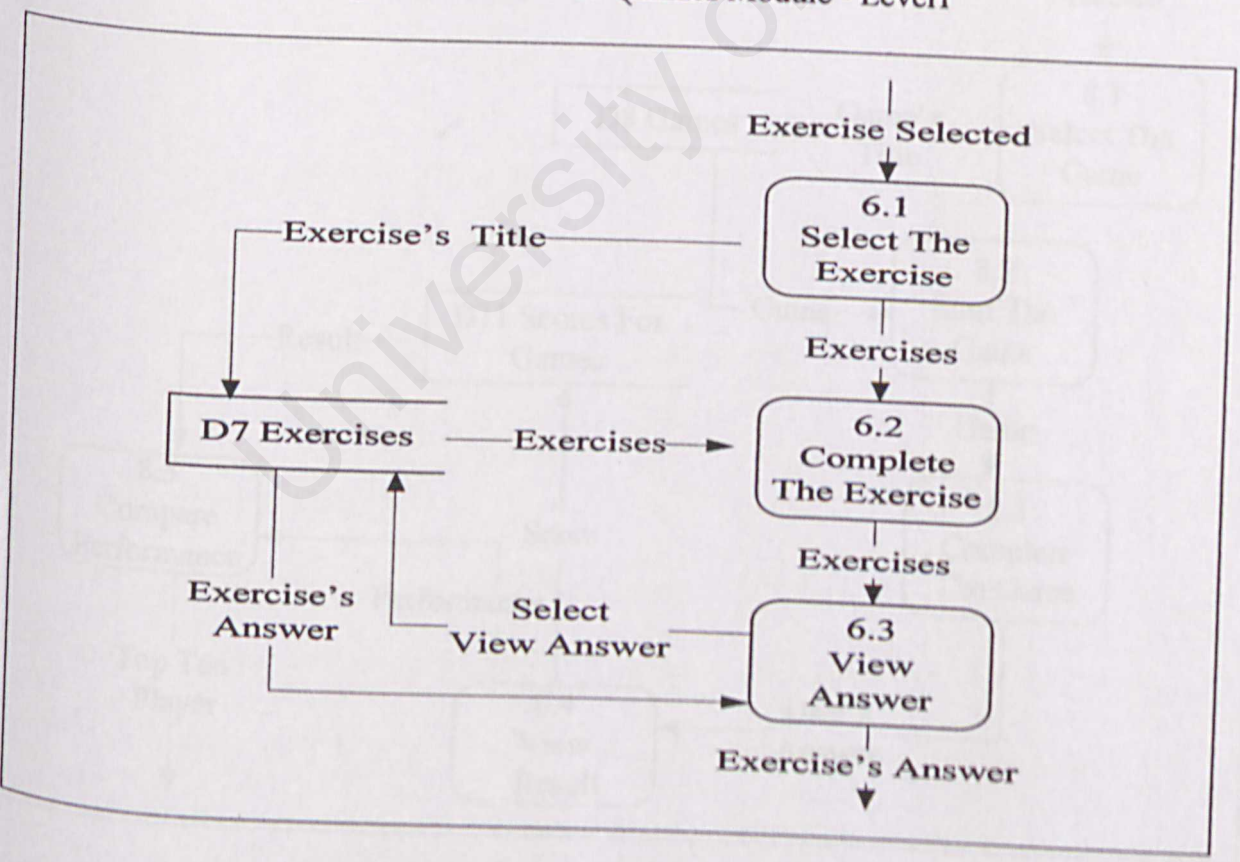

Figure 5.17: DFD of Exercises Module - Level1

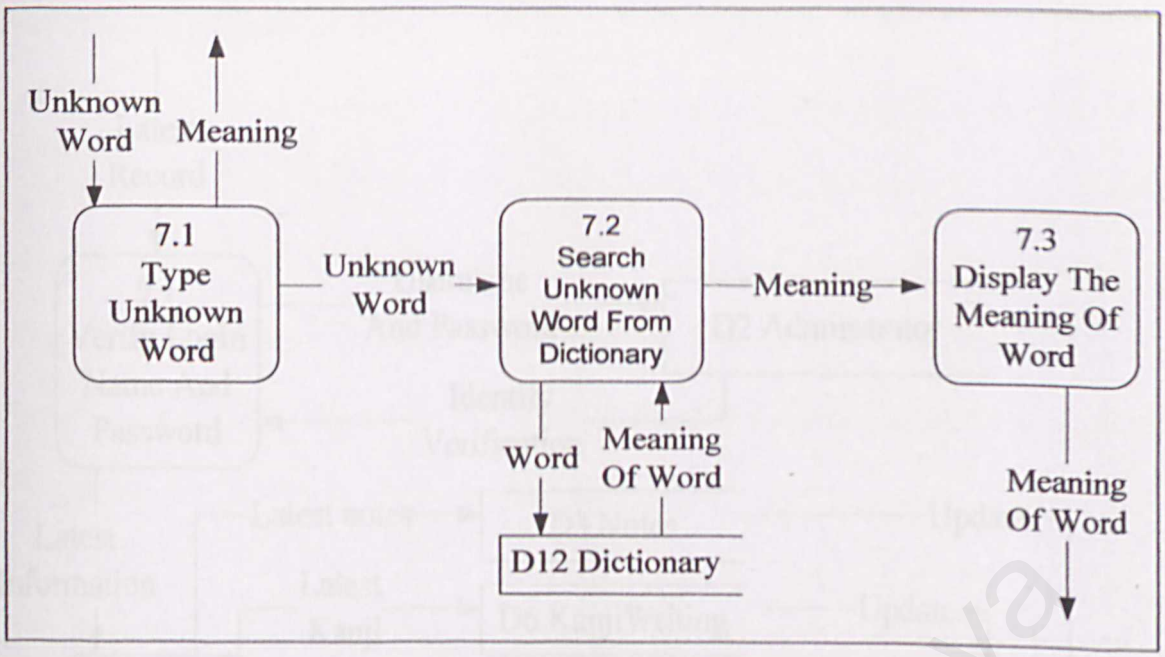

Figure 5.18: DFD of Dictionary Module - Level1

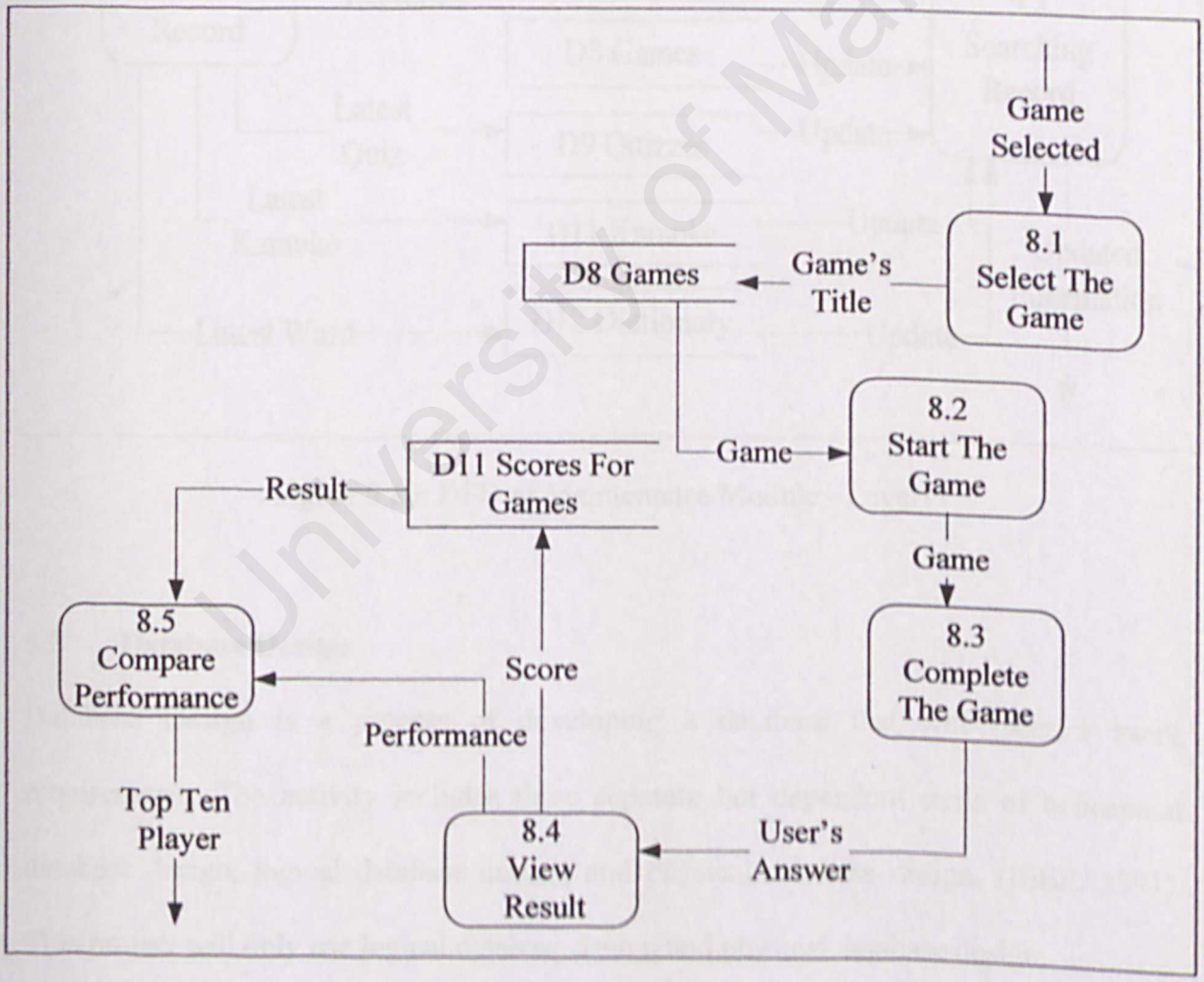

Figure 5.19: DFD of Games Module - Level1

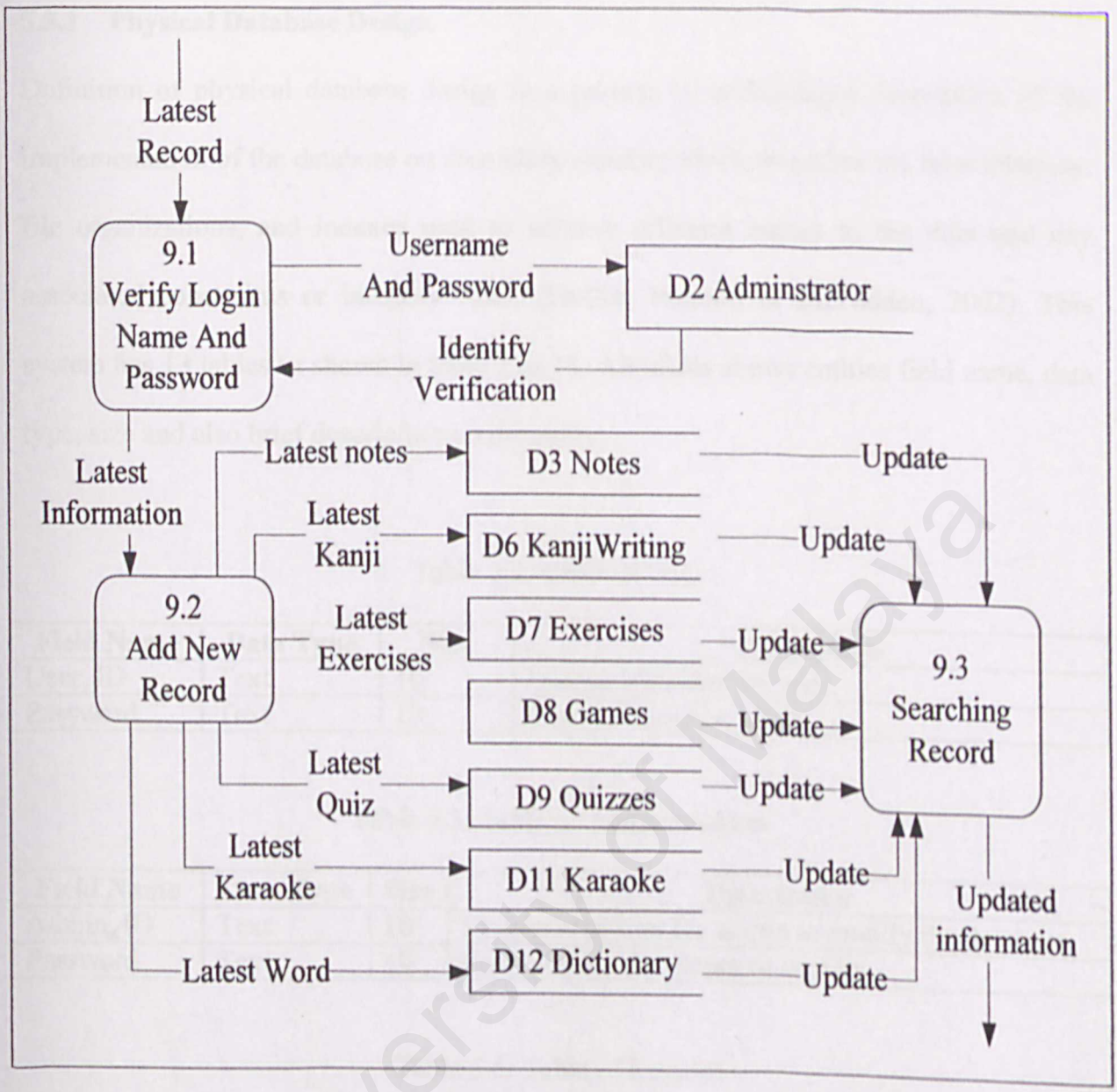

Figure 5.20: DFD of Maintenance Module - Level1

5.3 Database Design

Database Design is a process of developing a database that will meet a user's requirements. The activity includes three separate but dependent steps of conceptual database design, logical database design, and physical database design. (IEEE, 1991). This project will only use logical database design and physical database design.

5.3.1 Physical Database Design

Definition of physical database design is a process of producing a description of the implementation of the database on secondary storage, which describes the base relations, file organizations, and indexes used to achieve efficient access to the data and any associated constraints or integrity rules. (Hoffer, Prescott & McFadden, 2002). This system has 13 tables as shown in table 2 to 14. All tables shows entities field name, data type, size and also brief description on the entity.

Table 5.2: Table of User

Field Name	Data Type	Size	Description
User_ID	Text	10	Unique identifier for user.
Password	Text	10	Password user to log in.

Table 5.3: Table of Administration

Field Name	Data Type	Size	Description
Admin_ID	Text	10	Unique identifier for admin to modify the database.
Password	Text	10	Password for developer to sign in.

Table 5.4: Table of Lessons

Field Name	Data Type	Size	Description
Lessons_ID	Autonumber	Integer	Unique identifier for notes.
English_Titles	Text	100	Title of lesson in English.
Malay_Titles	Text	100	Title of lesson in Malay.
Japanese_Titles	Text	100	Title of lesson in Japanese.
English_Lessons	Text	100	Path to access notes for English version.
Malay_Lessons	Text	100	Path to access notes for Malay version.
Japanese_Lessons	Text	100	Path to access notes for Japanese version.
English_Translation	Text	100	Path to access English translation of notes which store in word file.
Malay_Translation	Text	100	Path to access Malay translation of notes which store in word file.

Table 5.5: Table of HiraganaWriting

Field Name	Data Type	Size	Description
Hiragana_ID	Autonumber	Integer	Identified each Hiragana.
Hiragana_Text	Text	3	Ro-manji
Hiragana_Sound	Text	100	Path to access sound file of each Hiragana.
Hiragana_Animate	Text	100	Path to access flash file of each Hiragana.

Table 5.6: Table of KatakanaWriting

Field Name	Data Type	Size	Description
Katakana_ID	Autonumber	Integer	Identified each Katakana.
Katakana_Text	Text	3	Ro-manji
Katakana_Sound	Text	100	Path to access sound file of each Katakana.
Katakana_Animate	Text	100	Path to access flash file of each Katakana.

Table 5.7: Table of KanjiWriting.

Field Name	Data Type	Size	Description
Kanji_ID	Autonumber	Integer	Identified each Kanji.
Kanji_Text	Text	50	Ro-manji
Kanji_Sound	Text	100	Path to access sound file of each Kanji.
Kanji_Animate	Text	100	Path to access flash file of each Kanji.
Kanji_Furi	Text	50	Furigana.

Table 5.8: Table of Exercises

Field Name	Data Type	Size	Description
Exercise_ID	Autonumber	Integer	Unique identifier for each exercise.
Exercise_Source	Number	Integer	Path to access each exercise.

Table 5.9: Table of Quizzes

Field Name	Data Type	Size	Description
Quiz_ID	Autonumber	Integer	Unique identifier for each game.
Quiz_Source	Number	Integer	Path to access each game.

Table 5.10: Table of Games

Field Name	Data Type	Size	Description
Game_ID	Autonumber	Integer	Unique identifier for each game.
Game_Source	Number	Integer	Path to access each game.

Table 5.11: Table of Scores for Quizzes

Field Name	Data Type	Size	Description
User_ID	Text	10	Identify each user.
Quiz_ID	Autonumber	Integer	Unique identifier for each exercise.
Quiz_Score	Number	Integer	Score obtained.
Quiz_Date	Date/Time	General Date	Date user does the exercise.
Quiz_Timer	Number	Integer	Time used by user to complete an exercise.

Table 5.12: Table of Scores for Games

Field Name	Data Type	Size	Description
User_ID	Text	10	Identify each user.
Game_ID	Autonumber	Integer	Path to access each game.
Game_Score	Number	Integer	Score obtained.
Game_Date	Date/Time	General Date	Date user does the exercise.
Game_Timer	Number	Integer	Time used by user to complete an exercise.

Table 5.13: Table of Karaoke

Field Name	Data Type	Size	Description
Clip_ID	Autonumber	Integer	Name of clips file.
Clip_File	Text	100	Path to access Audio clips of the karaoke.

Table 5.14: Table of Dictionary

Field Name	Data Type	Size	Description
Word_ID	Autonumber	Integer	Unique identifier for words.
Word	Text	50	Unknown word inserted by user.
English_Mean	Text	100	Path to access the meaning in English version.
Malay_Mean	Text	100	Path to access the meaning in Malay version.

5.3.2 Logical Database Design

Figure 5.21 shows the logical database design for the system.

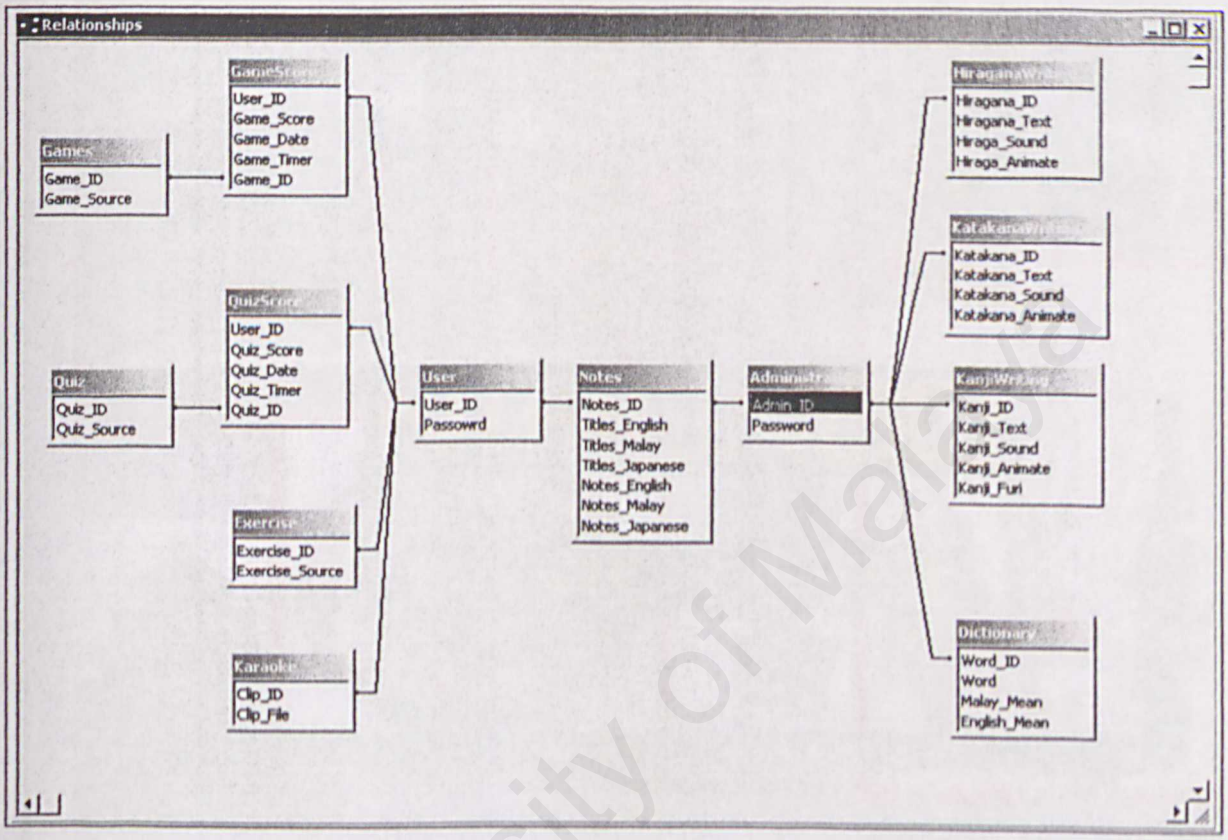

Figure 5.21: Logical Database Design

5.4 User Interface Design

One of the most important aspects for NihonGo process is the user interface design as it describes how the software interacts with the user. A good user interface should have consistency, effective, interactivity between the user and the system, easy to understand information display and user friendly data input. The follow are the user interfaces for this system:

Figure 5.22: Login Form for User

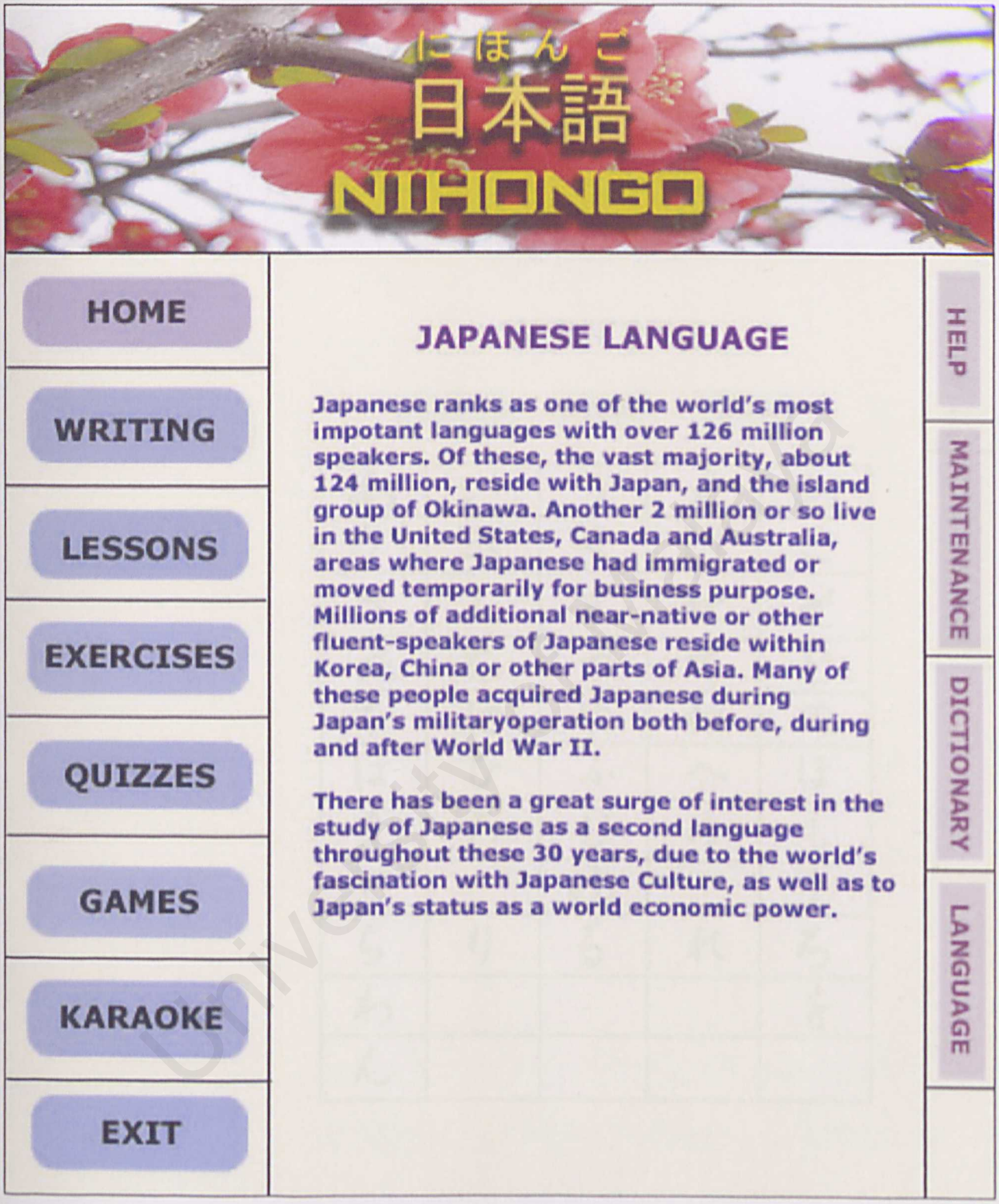

Figure 5.23: Home Form for User

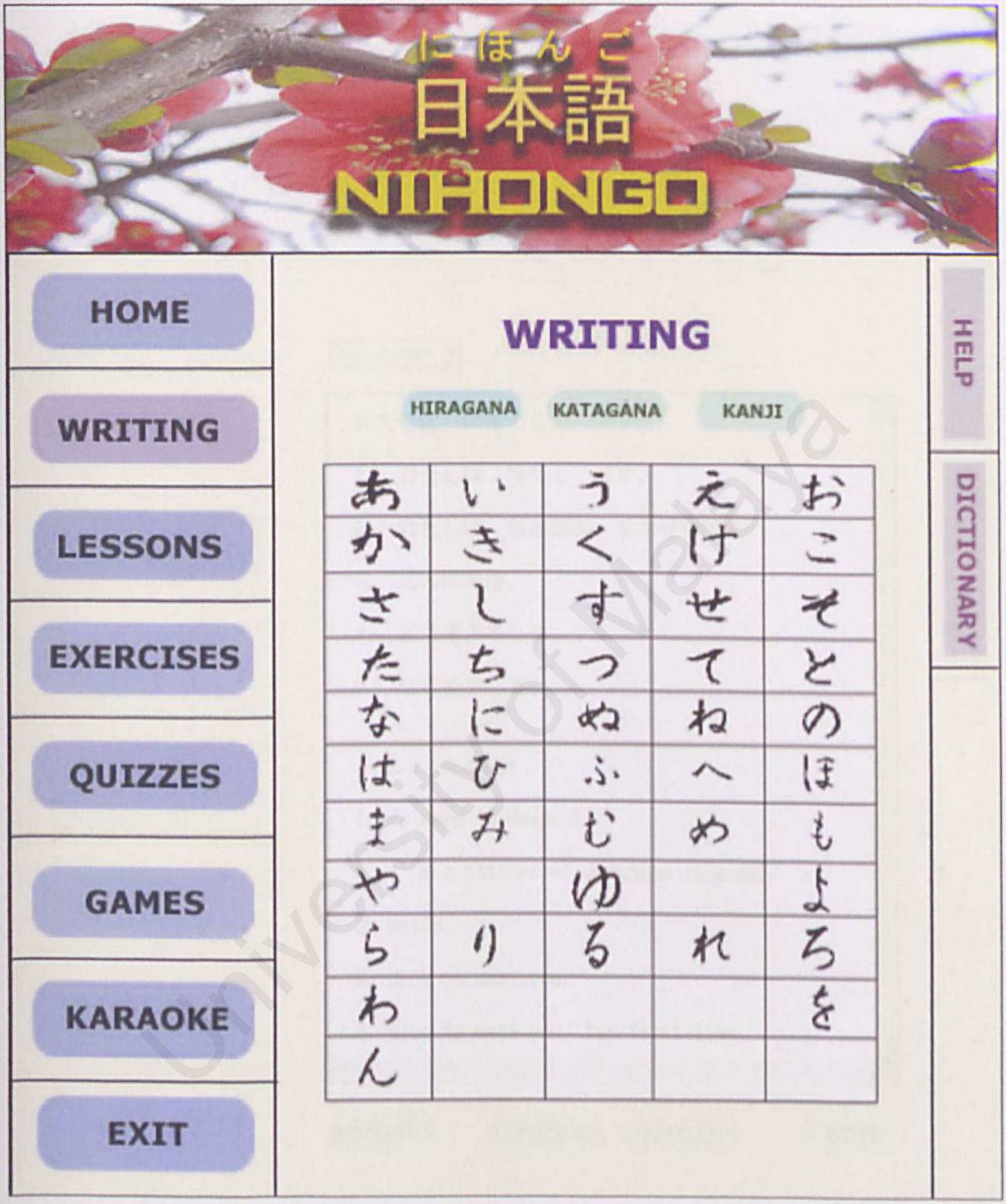

Figure 5.24: Writings Module

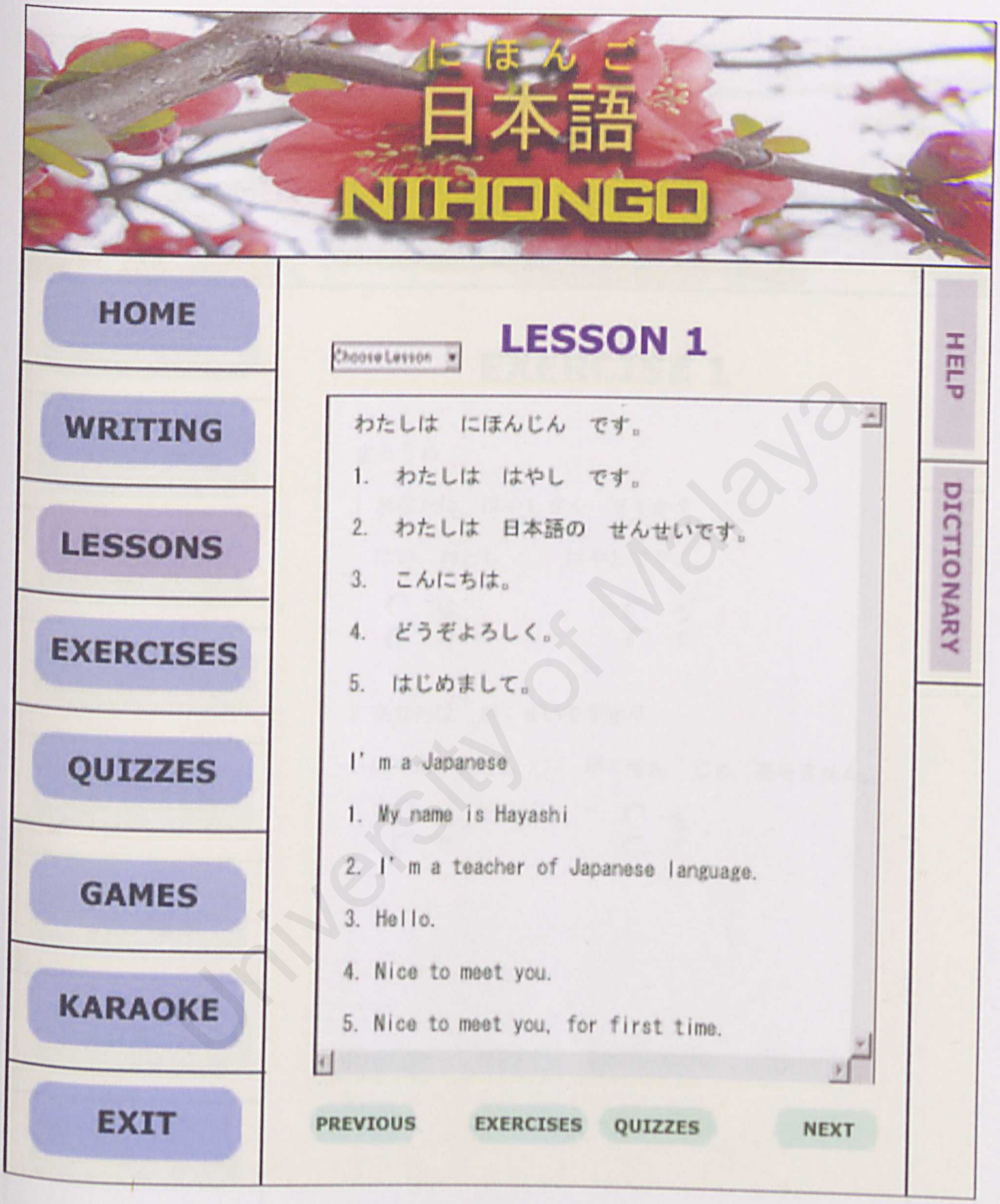

Figure 5.25: Lessons Module

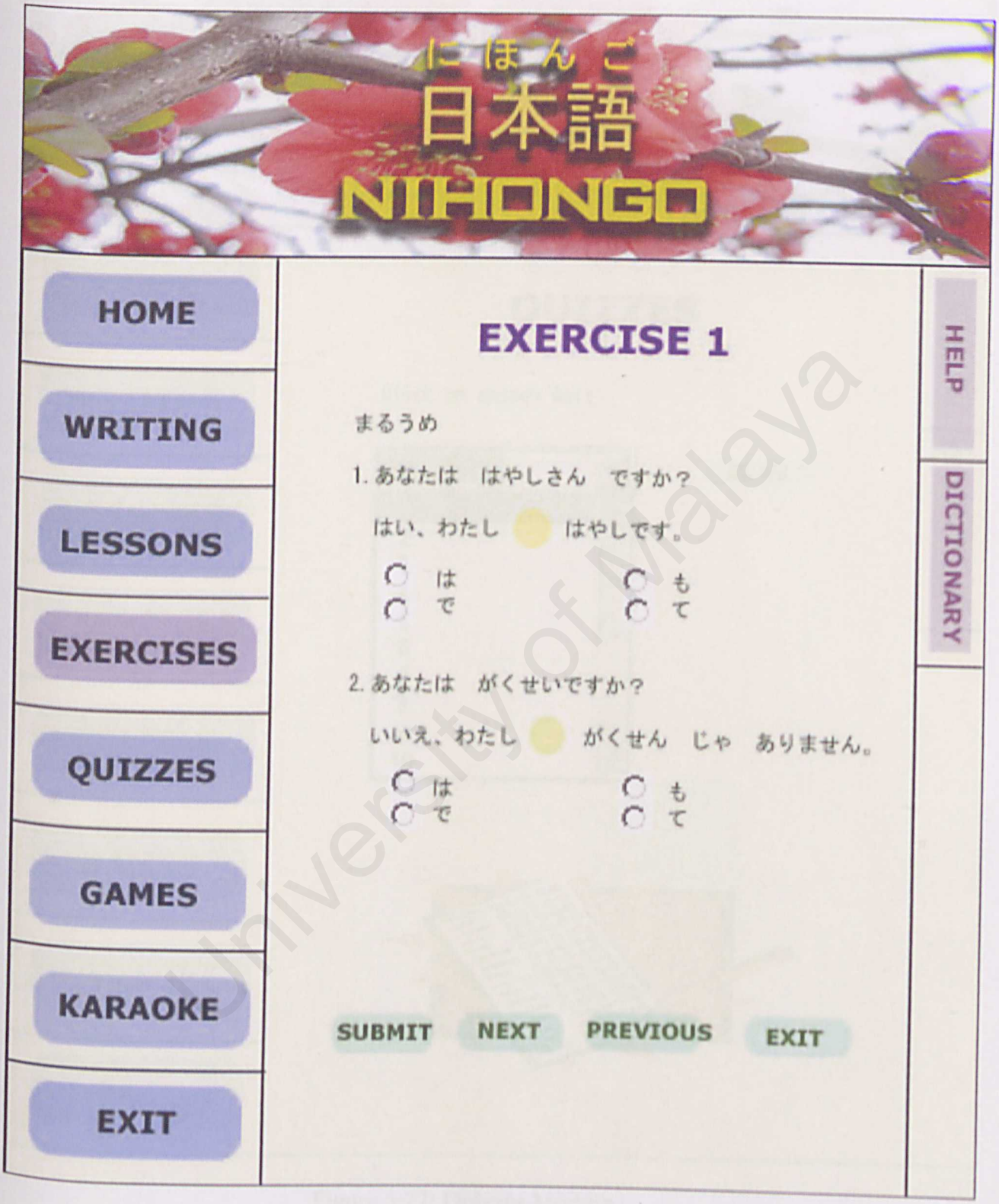

Figure 5.26: Exercises Module

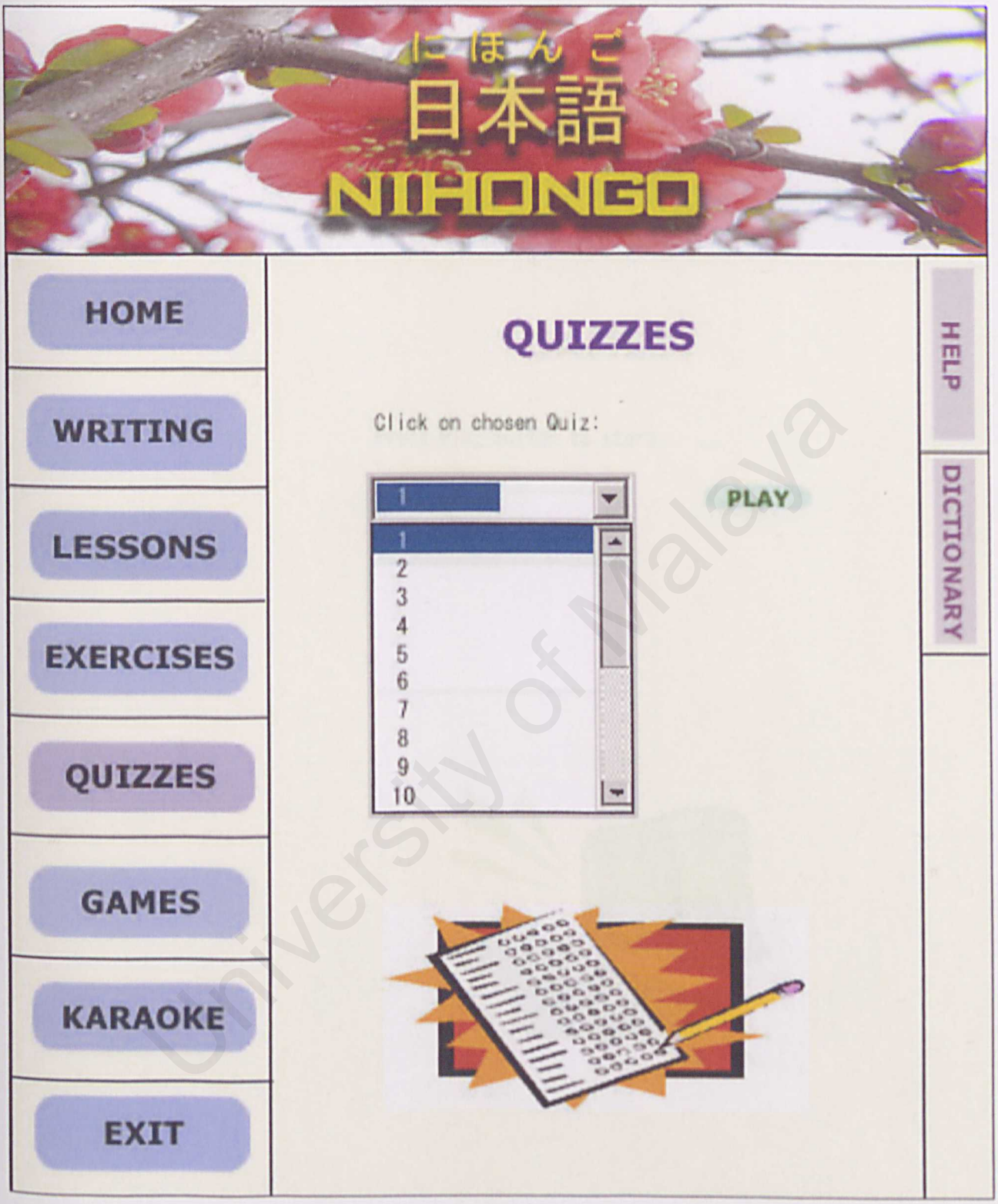

Figure 5.27: Quizzes Module

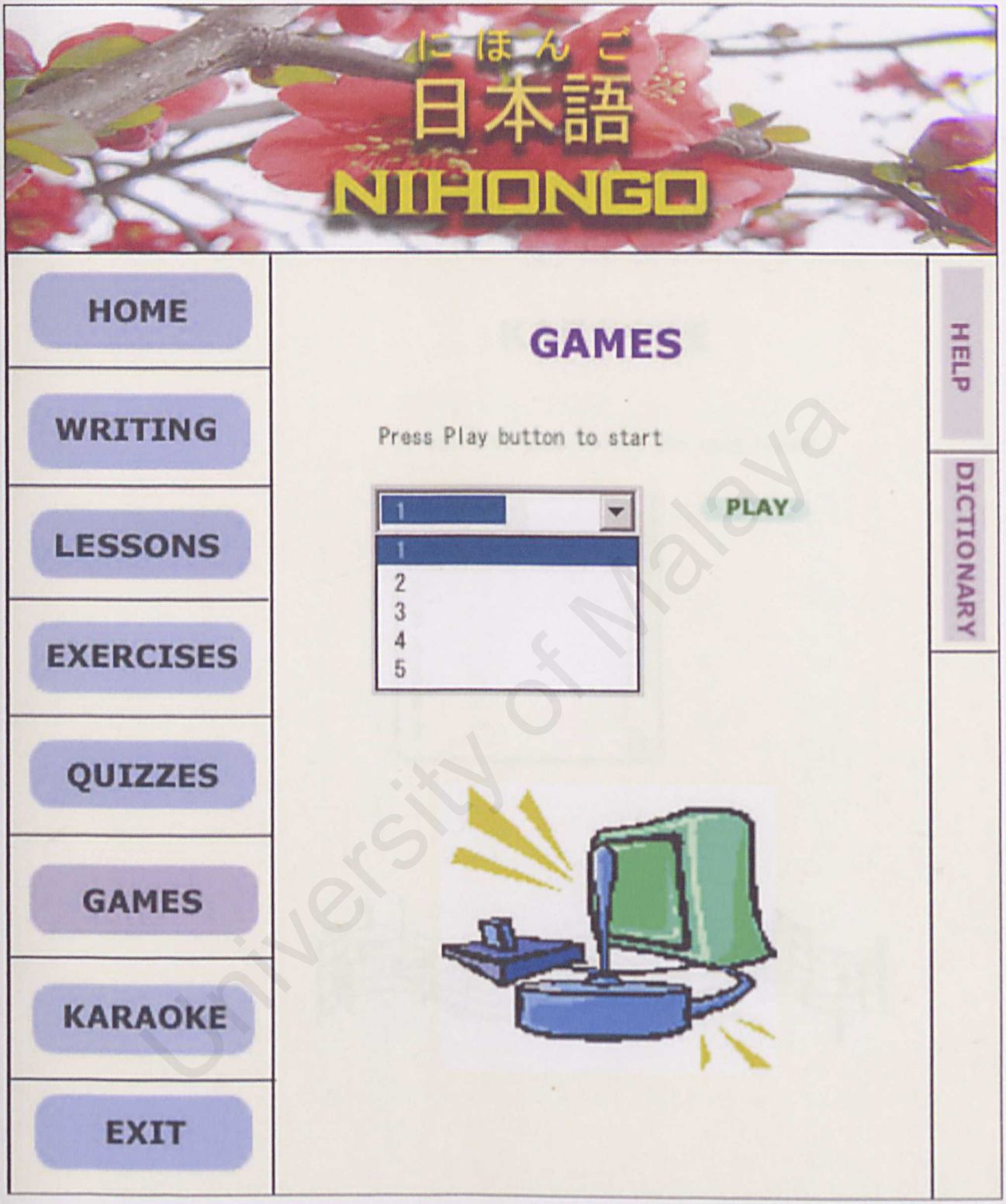

Figure 5.28: Games Module

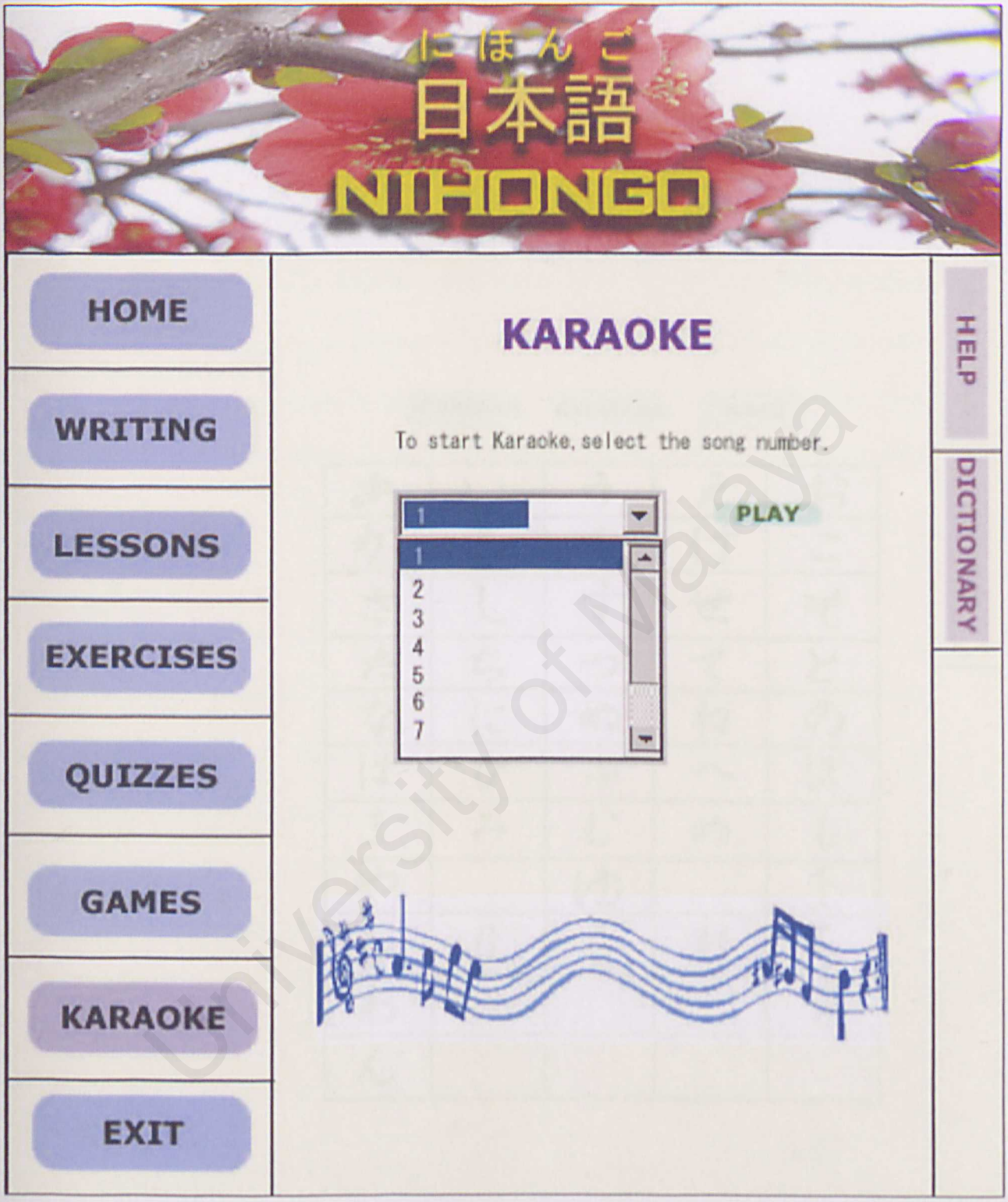

Figure 5.29: Karaoke Module

Figure 5.30: Writings Module (Malay version)

Chapter 6: System Implementation

6.1 Introduction

System implementation is a process commences after the design phase. System requirements and designs are interpreted into software code. The detailed design provides information for the programmers to implement the code. System implementation involves integration of individual codes from several modules into complete software as well. This phase will describe the initial and revised process design implementation in real world.

6.2 Development Environment

The development tools used to develop the system consist of hardware and software tools. Appropriate software and hardware speed up the system development according to schedule. The tools are described as below:

6.2.1 Software Requirements

The software tools used to develop NihonGo are listed in Table 6.1:

Table 6.1: The Software Tools for Development

Software/Tools	Purpose	Description
Windows XP Professional	System Requirement	Operating System (OS)
Microsoft Access 2000	Database	Build the Database
Microsoft Visual Basic.NET (VB.NET)	Coding	System Coding
Macromedia Dreamweaver MX 2004	HTML	Create HTML Notes
Adobe Photoshop CS	Interface Design and Contents Design	Image Design and Creation
Microsoft Word 2003	Documentation	Writing Documentation

6.2.2 Hardware Requirements

The hardware tools used to develop NihonGo are listed as below:

- Intel Pentium IV 1.6 GHz
- 512MB of RAM
- 40 GB Hard Disk
- 52X CD-ROM
- Speaker
- Scanner
- Other desktop peripherals like monitor, mouse and keyboard

6.3 System Development

This phase focuses on the technology and development tools usage analysis during the NihonGo development.

6.3.1 Database Development

Based on the logical data model for NihonGo created during the System Design phase, Microsoft Access 2003 is used as the Database Management System (DBMS). It incorporates visual user interface for designing and working with databases.

Firstly, the database development started by creating an empty database called Japan. Database is developed according to the database design in Chapter 4 but there are additional fields added. Data structure of each table is declared, table's primary key is set and the data retrieval is based on the SQL statement in VB.NET.

6.3.2 User Interface Development

The interface is developed best suited for high color screens 1024 x 768 resolutions. By having Graphical User Interfaces (GUIs), it supports the interaction using a mouse and keyboard to communicate with the user. In this system, Adobe Photoshop CS and Microsoft Visual Basic.NET are used for developing the system's interface.

6.3.3 Application Development

The NihonGo application development uses programming code in VB.NET. Standard principles are employed in coding method to ensure the system maintainability, readability and understandability. The principles used in the NihonGo development are as the following.

(a) Choose the standard meaningful procedure, variable and parameter names to avoid the excessive use of comments. The coding is shown in Figure 6.1.

```
Dim Chap() As Integer
Dim CountChap As Integer

Dim SpFlags As SpeechLib.SpeechVoiceSpeakFlags
Dim Voice As SpeechLib.SpVoice
Dim monAgent As AgentObjects.Agent
Dim monCharacter As AgentObjects.IAgentCtlCharacterEx
```

Figure 6.1: The Variable Declaration

(b) The code of database connection

```
conn = New ADODB.Connection
conn.Provider = "MICROSOFT.JET.OLEDB.4.0;Data source ='" &
Application.StartupPath & "\japan.mdb" & "'; Jet OLEDB:database
password=WEK020094"
conn.Open(Application.StartupPath & "\japan.mdb")
```

Figure 6.2: The Code of Database Connection

(c) Once establishing a database connection, SQL (Structured Query Language) statements are used to retrieve data from the databases. The following SQL statements retrieve user's identification from table "member" in database.

```
sql = "SELECT * FROM member WHERE Name='" &
Trim(Me.txtName.Text) & "'"
sql = sql & "AND ID='" & Trim(Me.txtPassword.Text) & "'"
```

Figure 6.3: The Code of SQL Statements

(d) In order to handle error, exception handling is shown in Figure 6.5

```
Try
 If (chkEnglish.Checked = False And chkJapan.Checked = False)
And ( _
  chkMalay.Checked = False And chkJapan.Checked = False) Then
        MsgBox("Please select a language")
Catch ex As Exception
        MsgBox(ex.Message)
End Try
```

Figure 6.4: The Code of Exception Handling

6.4 System Documentation

System Documentation is essential for developer to keep track of the project progress. The documentation describes the steps to perform a specific phase and the implementation of the system to the final acceptance test plan.

Therefore, user manual acts as a guide to users to use the program. It provides instructions on system installation, the procedure to operate the system and explanations.

Chapter 7: System Testing

7.1 Introduction

System testing is the process of executing a software system to determine whether it matches its specification and executes in intended environment. Software testing is often used in association with the terms verification and validation. Verification is the checking or testing of system, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques, such as reviews, analysis, inspections and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

There are very few modules written earlier performing correctly and also, the requirements kept changing during the development phase. The modules must be rewritten comprehensively to ensure that it is executed properly and well performed. System testing is to ensure the entire modules in the system can be executed correctly. Testing is carried out throughout software development instead at the end. Therefore, testing is a very important phase which has to be carried out well before the system is delivered to end-user. NihonGo is mainly tested for:

- Reliability: NihonGo operates without crashing, hanging or other run-time errors.
- Functionality: NihonGo meets all requirements established.
- Performance: NihonGo is able to respond in a timely manner.
- System Performance: NihonGo will continue to perform correctly in a timely manner when it is subjected to production load.

The other objective of testing is to ensure operational reliability by uncovering defects in the system. This objective is achieved by deliberately designing inputs of data and rigorously testing the system against these. The defects uncovered include:

- Logic errors,
- Coding errors,
- Technical language syntax errors, and
- Database integrity errors.

7.2 Testing Techniques Practiced

This section focuses on the testing techniques practiced throughout the application testing phase. In NihonGo application we chose to use ad-hoc testing, black-box testing and white box testing techniques.

7.2.1 Ad-hoc testing

Ad hoc testing is an attempt to break the program or make it fail with trying it with whatever input comes to mind.

7.2.2 Black box testing

Black box testing or functional testing is used to check outputs of the application, given certain inputs, conform to the functional specification of the program. The term black box indicates that the internal implementation of the program being executed is not examined by the tester. It focuses on the most important aspects of a module in the term of how

well the module meets its specification. Listed below are few approaches applied for black box testing technique:

- Error guessing

  This approach is similar to 'ad hoc testing', where the tester will attempt to input any type of test cases that came across his/her mind or a pre-planned test case.

- Boundary testing

  This approach include boundary of equivalent classes where the coverage of test cases will involve inside the boundary, on the boundary and outside the boundary.

- Module interface testing

  In this approach, each value within the interface is assured as correct as they related to the module that calls them. This means that specific calls in the calling module are tested whether they are in the right sequence.

7.2.3 White box testing

White box testing is used to check that the outputs of the application, given certain inputs, conform to the functional specification of the program. Unlike black box testing, white box testing is the type of testing that deals directly with the structure of the code within a module or a code segment. The term white box indicates that the internal implementation of the program being executed is examined by the tester. Listed below are few approaches applied for white box testing:

- Branch node coverage

  Each and every branch of every possible direction is taken at least once.

- Compound condition coverage

  When multiple conditional present in the code, every possible combination is tested based on a truth table.

- Basis path testing

  Each independent path through the code is tested thoroughly.

- Data flow testing

  This approach is to discover anomalies such as variables, which are not used but are not declared.

- Loop testing

  This approach related to test single loop as WHILE, FOR, IF ... ELSE, concatenated loops or sequence loops and also nested loops.

7.3 Testing the System

A software module is exposed to testing both in development phase and throughout the test and integration phase. Development phase comprise each function of the module is tested thoroughly upon completion of the module. The main difference between testing conducted in development phase and the test and integration phase is that, during the development phase, errors are found and fixed. While in the test and integration phase, if a failure present, it will be recorded and the failed module will returned to the development team along with an account of failures experienced. NihonGo had

undergoes three stages of testing before it was considered a complete system. They are Unit Test, Integration Test and System Test.

7.3.1 Unit Testing

Unit testing is the most basic level of code testing. It has two major objectives: to verify that the application software components code works according to its specifications, and to validate the programs logic. Unit testing accomplishes these objectives by pinpointing differences between what an application software component does and what it is supposed to do. The advantage of unit testing is that it permits the testing and debugging of small units of code, thereby providing a better way to manage the integration of units into larger components.

For NihonGo, unit testing is done subsequent to coding phase. After the source codes of a module has been developed, reviewed and verified for the correct syntax, unit-testing case was designed. The module was tested to ensure that it operates correctly.

7.3.2 Integration Testing

Integration testing, in which progressively larger groups of tested software modules corresponding to elements of the architectural design are integrated and tested until the software works as a whole. Throughout integration testing, test the combined individual and unit-based pieces of software of a complete system. Integration testing includes the integration of modules into programs, programs into subsystems, and subsystems into the overall system. This testing event uncovers errors that occur in the interactions and interfaces between units, which cannot be found during unit testing.

The major objectives of integration testing for NihonGo are to verify that interfaces between application software components function properly and interfaces between the application and its users function properly.

7.3.3 System Testing

In system testing all modules are integrated to the overall product and tested to show that all requirements are met. Testing the overall system is exceptionally varied from unit and integration testing. System testing evaluates the functionality and performance of the system. It encompasses testing for performance, stress, security, configuration sensitivity, usability, data integrity, startup and recovery. NihonGo system is tested whether it is functioning properly and ensures all requirements are met. The following system testing was carried out:

1. **Functional Test**

   The objective of functional test phase is to ensure that each element of the application meets the functional requirements of the software as outlined in the system design document. Effective function tests have a high probability of detecting a fault. NihonGo uses guidelines as listed below for functional test:

   - Know the expected actions and output.
   - Test for both valid and invalid input.
   - Not modify the system to justify with testing.

2. **Performance Test**

The purpose of application performance testing is to determine if the system can handle the anticipated amount of work in required time. The types of performance test carried out for NihonGo are timing tests and security tests.

3. **Acceptance Test**

Acceptance test is executed to ensure that the system operates in the manner expected, and any supporting material such as procedures, forms, etc. are accurate and suitable for the purpose intended. It is high level testing, ensuring that there are no gaps in functionality.

Chapter 8 System Evaluation and Conclusion

8.1 Problem Encountered and Solutions

- Text-To-Speech Synthesizer (TTS)

  This is a very advanced technology for us and due to the lack of knowledge; we are unable to commence work at first. However, with the help of our supervisor and senior, we managed to overcome this problem by using the Microsoft Agent Control 2.0 for the Japanese and English TTS, whereas for Malay TTS, we implement with MBrola.

- Choosing the software development kits

  There are many software development tools to develop this system. With the guidance and advices from our supervisor, finally we decided to choose Visual Studio.Net because of its easiness in using its features and the strong library functions.

- Determine the scope of the system

  Since there is an abundance of Japanese reference out there, it is really difficult to select the scope of the notes. Therefore, we took our supervisor's advice by following the lessons that she had taught us in the extra class and follow her notes from Chapter 1 to Chapter 10.

- Time Constraint

  During the design phase, we only realized it is hard to set the notes and questions for each chapter because it is hard for us to understand the Japanese which is more difficult for the Chapter 10 onwards. Besides that, due to the

inexperienced and insufficient time, we spend much time in programming. Thus, we seek help from other friends and read a lot for the relevant examples and find resources from internet.

8.2 System Strength

- Simple and user-friendly interfaces

  The system provides simple, consistent and data integrity interface. The user can easily familiarize or get accustomed with the system's concept based style. User may need just minimal knowledge to use this system where an action is just a click away.

- Attractive graphics and interfaces

  The eye-catching graphics, with beautiful background pictures certainly will attract the user to use it.

- Text-To-Speech (TTS)

  TTS allows applications to stream text from virtually any source for conversion to an audio format. Normally, we can save time and cost without having to pre-record information for storage as sound files. Besides that, it is a good way to present dynamic data using TTS because some speech applications can be deployed.

- Interesting games

  The games are developed in an interesting and interactive style. Moreover, user can relax and have fun in the games.

- Interactive learning through games and karaoke

  Through these two modules, user can learn vocabulary and speaking simultaneously. Games module requires user's vocabulary memorization. Karaoke module improves user the vocabulary intonation and speaking.

- Bilingual (Malay and English)

  This system is available in bilingual. Thus, user can choose which language that they most familiar with and hence language is not a hinder towards the curve of learning process.

- Dictionaries

  The dictionary database can easily check and explain the meaning in bilingual.

8.3 System Limitations

- Standalone system

  User needs to install the system into their computer before using it. However, the system can support multi-user environment.

- Have to install .Net Framework 1.1

  In order to run the system smoothly, user need to install .Net Framework 1.1. Moreover, user needs to install Microsoft Agent Control, Microsoft Speech Object Library and Speech SAPI 4.0 to run the TTS correctly.

- Language selection is done at the Main menu only

  User can only choose the language at the start menu but not inside each module. This is because frequent language changes in each module will confuse the database and will increase the software runtime in term of performance.

- Password is hashed but not encrypted.
- Malay TTS is not part of Microsoft Speech Object Library (external sources). Thus, Microsoft Agent Control 2.0 does not have character display for Malay TTS.

8.4 Future Enhancement

- The speed control of flash in writing module.
- The creation of Microsoft Agent Control 2.0 character in Malay TTS.
- Increase bank question, notes and karaoke songs.
- Password encryption.
- The Kanji in Writing Module needs animated graphics.

8.5 Knowledge and Experience Gained

- Project planning skills

  In order to accomplish the project on time, a good project planning is required. Therefore, we plan a schedule and phases since the project commence until the system is completely developed and delivered. This may reduce project failure risk and ensure that the project meets the objectives.

- Mastery of VB.Net and related software

  By developing this system, we become more familiar with VB.NET programming environment, and Microsoft Access. Besides that, we also gained experience from using Adobe Photoshop CS and Macromedia Fireworks MX, to edit a lot of graphics. The web page is developed using Macromedia Dreamweaver MX.

- Hands on Experience

  We had gained experiences in developing this system. The concepts and theories which we had learned in project management, software engineering, object-oriented concept and so forth are very useful and applied during the Software Development Life Cycle (SDLC) process.

- Finding Information and solutions from internet

  Internet is a mega database where you may find a lot of information. We often referred to the search engines to find solutions or post the questions in the forums to discuss and get solutions from the experts.

- Understand better regarding Japanese

  We managed to learn all the hiragana, katakana and some basic Japanese words. Besides that we are also exposed to Japanese culture.

8.6 Conclusions

In a nutshell, NihonGo has been successfully developed and achieved its objective together with the strength and limitations stated earlier. This is an attractive and interactive learning package which makes the learning process more efficient. However, there is still needs for improvement. By taking up the necessary future enhancement steps, all the limitations can be overcame.

Last but not least, this is a very good practical testing on undergraduate capabilities in handling and developing project. It also works as a stepping stone for them to apply the gained knowledge and polish their skills before expose to

the real working environment. It is a challenge, to develop this computer aided learning package; a lot of efforts, research and time are needed as to make it perfect.

References

(Access 2003 Product Overview, 2003) Access 2003 Product Overview.
http://www.microsoft.com/office/access/prodinfo/overview.mspx.
Date updated: June 9, 2003. Microsoft Corporation. Date Accessed: September 7, 2004.

(Adobe Photoshop CS, 2004) Adobe Photoshop CS.
http://www.adobe.com/products/photoshop/main.html
Adobe. Date Accessed: September 7, 2004.

(Ager, 2004) Ager, S. 2004. Japanese (Nihongo).
http://www.omniglot.com/writing/japanese.htm
Omniglot: Writing Systems. Date Accessed: September 28, 2004.

(Dampier, 2004) Dampier, D.A. 2004. Rapid Prototyping and Incremental Evolution Software.
http://www.dacs.dtic.mil/awareness/newsletters/technews2-1/evolution.html.
Software Tech News, National Defense University. Date Accessed: September 27, 2004.

(Declan Software: Japanese, 2004) Declan Software: Japanese
http://www.declan-software.com/japanese
Declan Software. Date Accessed: September 18, 2004.

(Euro Talk: Japanese, 2004) Euro Talk: Japanese
http://www.eurotalk.com/Library/fmpro?-db=web_prod.fp5&-lay=web&-format=/pages/product.html&productid=AMT5020&-find
Euro Talk Interactive. Date Accessed: August 28, 2004

(Features by Edition for Visual Basic 6, 2004) Features by Edition for Visual Basic 6.
http://msdn.microsoft.com/vbasic/previous/vb6/datasheet/feature.aspx.
Microsoft Corporation. Date Accessed: September 6, 2004.

(Features Overview for Visual Basic .NET 2003, 2004) Features Overview for Visual Basic .NET 2003.
http://msdn.microsoft.com/vbasic/productinfo/features/default.aspx.
Microsoft Corporation. Date Accessed: September 6, 2004.

(Hoag, 2002) Hoag, S. 2002. Deployment Changes in Visual Basic .NET.
http://msdn.microsoft.com/vbasic/default.aspx?pull=/library/en-us/dv_vstechart/html/vbtchdistributingapplicationscreatedwithvisualbasicnet.asp.
Date Updated: January, 2002. Microsoft Corporation. Date Accessed: September 23, 2004.

(Hoffer, Prescott & McFadden, 2002). Hoffer, J.A., Prescott, M.B., McFadden, F.R. 2002. *Modern Database Management.* 6th Edition. New Jersey, Prentice-Hall Inc.

(IEEE, 1991) IEEE STD 1074. 1991. *IEEE Standard for Developing Life Cycle Processes.* New York, NY: Institute of Electrical and Electronics Engineers.

(IPC Editorial Board, 2000) IPC Editorial Board. 2002. *Basic Japanese Conversation.* Kuala Lumpur. Intellectual Publishing Co.

(Japanese Language: Where Do I Begin, 2004)
http://japanese.about.com/blbegin.htm
About.com. Date Accessed: September 12, 20004

(Kana-kun, 2004) Kana-kun.
http://www.kanjikit.net/kana-kun/info.htm
Kanji-kit.net. Date Accessed: September 13, 2004

(Macromedia Director MX 2004, 2004) Macromedia Director MX 2004.
http://www.macromedia.com/software/director/productinfo/overview/.
Macromedia. Date Accessed: September 8, 2004.

(Macromedia Flash MX 2004, 2004) Macromedia Flash MX 2004.
http://www.macromedia.com/software/flash/.
Macromedia. Date Accessed: September 8, 2004.

(Macromedia Flash MX and Director Comparison, 2004) Macromedia Flash MX and Director Comparison.
http://www.macromedia.com/software/director/resources/integration/flash/quick_comparison.html.
Macromedia. Date Accessed: September 8, 2004.

(Multimedia Japanese Learning Corner, 2004) Multimedia Japanese Learning Corner.
http://www.cuhk.hk/jas/jlcenter/jlcenter.htm.
Date updated: January 1, 1998. Department of Japanese Studies, Chinese Universities of Hong Kong. Date Accessed: September 28, 2004.

(Oblinger & Rush, 1997). Oblinger D. G. and Rush S. C. 1997.
http://www.cs.bris.ac.uk/~monadjem/cal11.pdf
Date Accessed: September 28, 2004.

(Schach, 2003) Schach, S.R. 2003. *Object Oriented & Classical Software Engineering.* 6th Edition. New York. McGraw-Hill.

(Sterzinger, 1998) Sterzinger, B.E. 1998. Japanese Writing Tutor Mirror Sites.
http://members.aol.com/writejapan. Date Accessed: September 28, 2004.

(SQL Server 2000 Product Overview, 2002) SQL Server 2000 Product Overview. *http://www.microsoft.com/sql/evaluation/overview/default.asp.* Date updated: December 31, 2002. Microsoft Corporation. Date Accessed: September 7, 2004.

(Takase, 2004) Takase, E. 2004. Japanese Writing System. *http://www.takase.com/Library/Background.htm.* Takase Studios, Inc. Date Accessed: September 28, 2004.

(Whitten, Bentley & Dittman, 2000) Whitten, J.L., Bentley, L.D. and Dittman K.C., 2004. *System Analysis and Design Methods.* 6th Edition. New York. MacGraw-Hill.

APPENDIX A USER MANUAL

TABLE OF CONTENT FOR USER MANUAL

Table of contents for user manual

List of figures for user manual

Chapter 1 Main Menu **1**

1.1 User's Login 1

1.2 Main Menu 2

Chapter 2 Writing Module **3**

2.1 Writing (Hiragana, Katakana, Kanji) 3

Chapter 3 Notes Module **5**

3.1 Notes' Lessons 5

3.2 Note's Information 7

3.3 Note's Dictionary 8

3.4 Using Help 10

Chapter 4 Exercises Module **11**

4.1 Taking Exercises 11

Chapter 5 Quizzes Module **12**

5.1 Taking Quiz According To Chapter 12

5.2 Performance Viewing 14

Chapter 6 Karaoke Module **16**

6.1 Playing Karaoke Songs 16

Chapter 7 Games Module **17**

7.1 Games: Who Wants To Be A Millionaire 17

7.2 Boulder Game 19

Chapter 8 Maintenance Module **20**

8.1 Administrator's Login 20

8.2 Notes Maintenance 21

8.3 Dictionary Maintenance 24

8.4 Exercise Maintenance 26

8.5 Quiz Maintenance 28

List of Figures for User Manual

Figure		Page
Figure A-1.1(a)	Log In	1
Figure A-1.2(a)	Fun Button Click	2
Figure A-1.2(b)	Learn Button Click	2
Figure A-2.1(a)	Choose Writing	3
Figure A-2.1(b)	Choose Writing Level	4
Figure A-2.1(c)	Character Animation	4
Figure A-3.1(a)	Notes' Lessons	5
Figure A-3.1(b)	Notes' Text-To-Speech	6
Figure A-3.2(a)	Japan Information	7
Figure A-3.3(a)	Notes' Dictionary	8
Figure A-3.3(b)	Notes' Search Result	9
Figure A-3.4(a)	Using Help	10
Figure A-3.4(b)	Help Menu	10
Figure A-4.1(a)	Taking Exercises	11
Figure A-5.1(a)	Quiz Selection	12
Figure A-5.1(b)	Start Taking Quizzes	13
Figure A-5.2(a)	Performance Viewing	14

Figure A-5.2(b)	Display Performance Graph	15
Figure A-6.1(a)	Playing Karaoke Songs	16
Figure A-7.1(a)	Game Choices	17
Figure A-7.1(b)	Who Want To Be A Millionaire	18
Figure A-7.2(a)	Boulder Game	19
Figure A-8.1(a)	Administrator Log In	20
Figure A-8.2.1(a)	Notes Maintenance	21
Figure A-8.2.2(a)	Notes' Search	22
Figure A-8.2.2(b)	Record Updated	23
Figure A-8.2.2(c)	Data Deletion	23
Figure A-8.3.1(a)	Add New Record	24
Figure A-8.3.2(a)	Search Record	25
Figure A-8.4.1(a)	Add Exercise Record	26
Figure A-8.4.2(a)	Search Exercise Record	27
Figure A-8.5.1(a)	Add Quiz Record	28
Figure A-8.5.2(a)	Search Quiz Record	29

Chapter 2 Writing Module

2.1 Writing (Hiragana, Katakana, Kanji)

1. In Figure A-2.1(a), click on any of the 3 button to choose the writing. Each button will have tool tip text in romanji.
2. In Figure A-2.1(b), click on the "level 1" button or "level 2" button to choose table.
3. Ciick on any character in the table.
4. Figure A-2.1(c) will pop up. An animation will show how to write the selected character stroke by stroke.
5. User can click on the speaker image button to hear the selected character pronunciation.
6. The meaning of the character combine with others character shown.

Figure A-2.1(a): Choose Writing

Figure A-2.1(b): Choose Writing Level

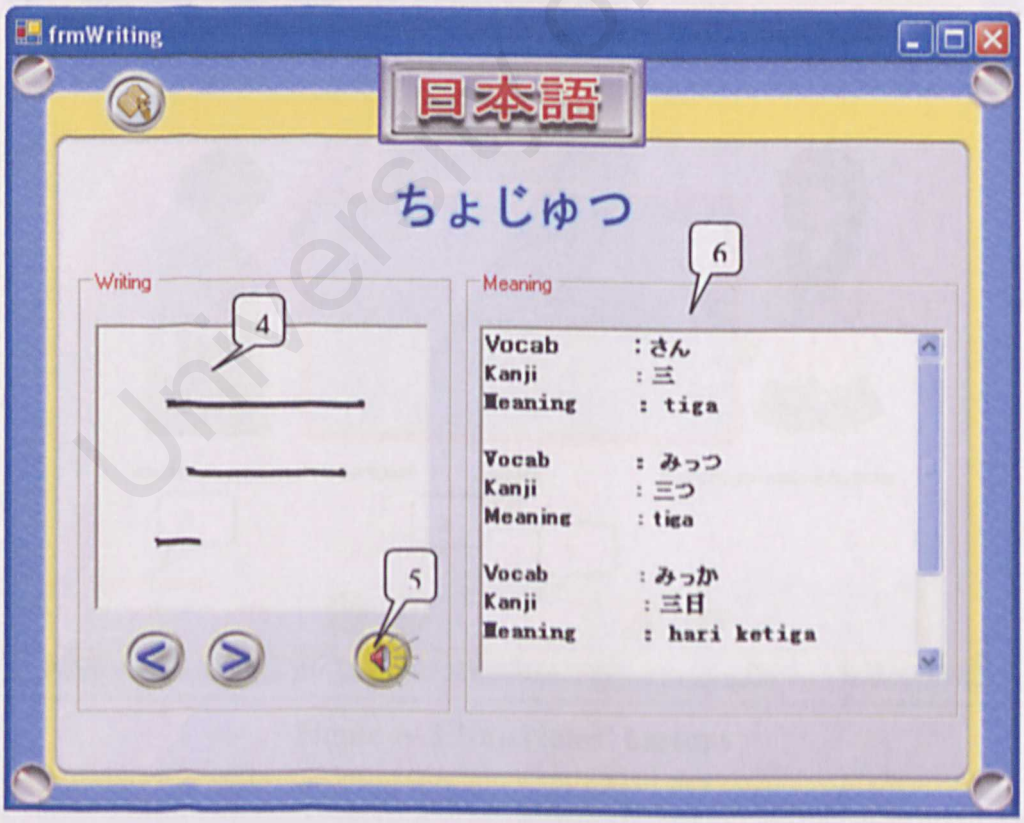

Figure A-2.1(c): Character Animation

Chapter 3 Notes Module

3.1 Notes' Lessons

1. Figure A-3.1 (a) shows the lessons provided for users. Users can click the combo box to choose the desired chapter.
2. Users can proceed to the next or previous lesson in the chapter.
3. To use the Text-To-Speech, users need to choose the spoken language.
4. Figure A-3.1 (b) shows the words pronunciations when the speaker button is pressed. A character will pop out and pronounce the grammar in required spoken language.

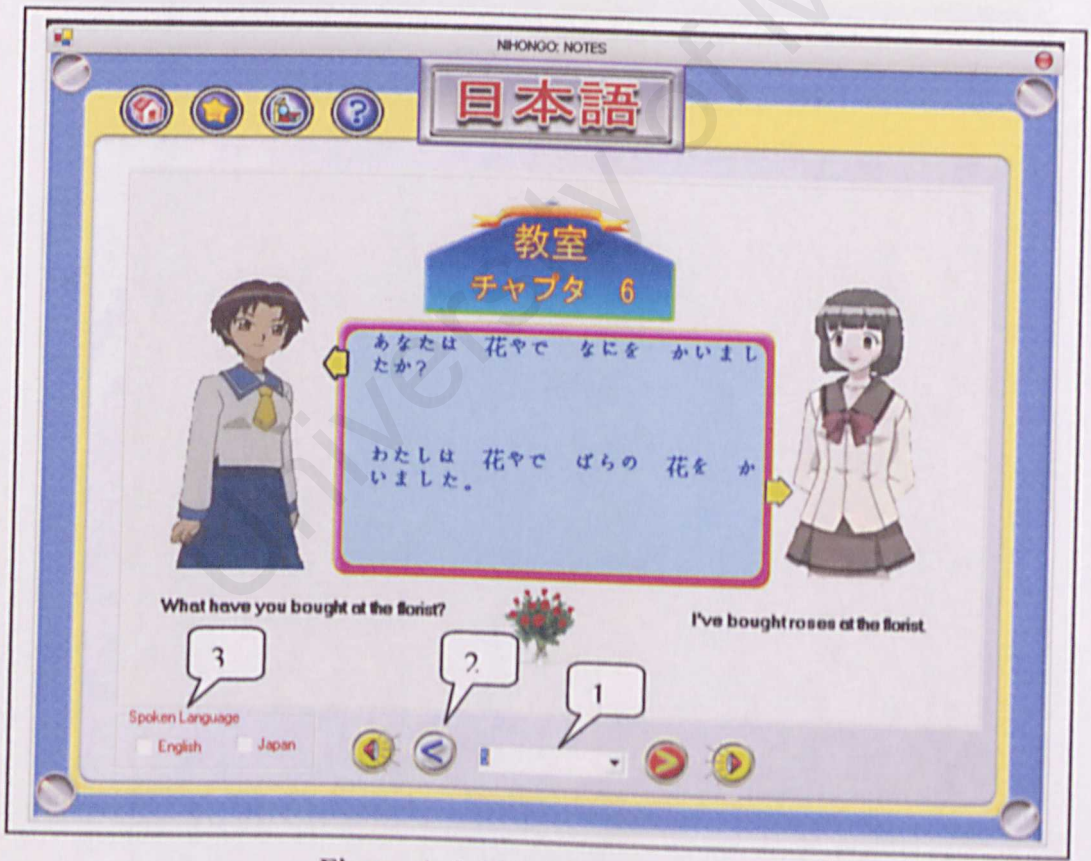

Figure A-3.1(a): Notes' Lessons

Figure A-3.1(b): Notes' Text-To-Speech

3.2. Note's Information

1. Figure A-3.2(a) shows the information on Japan's history, culture, festivals and language once the users click on the Information button.

2. Users can click to navigate the page to other page about Japan's information.

Figure A-3.2(a): Japan Information

3.3 Note's Dictionary

1. Figure A-3.3(a) is the page, which navigates to the dictionary splitter.
2. A list of similar words with the romanji appears once the user key in any alphabet or words.
3. Figure A-3.3(b) shows the desired meaning after the users' key in the romanji and click on the Search button.
4. To exit the dictionary, users have to click the Exit button.

Figure A-3.3(a): Notes' Dictionary

Figure A-3.3(b): Notes' Search Result

3.4 Using Help

1. In each module, a "Help" button is provided.
2. Click the "Help" button as shown in Figure A-3.4(a).
3. Figure A-3.4(b) shows the Help Menu. Hyperlinks are provided to navigate users to desired help.

Figure A-3.4(a): Using Help

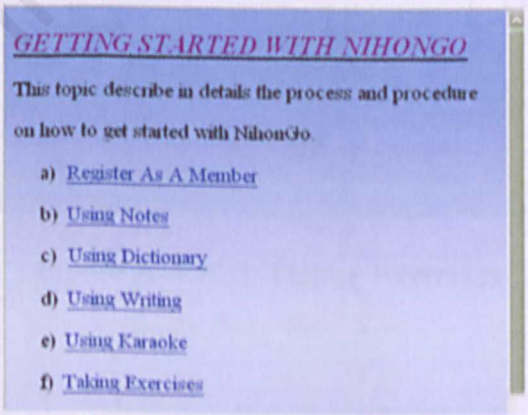

Figure A-3.4(b): Help Menu

Chapter 4 Exercises Module

4.1 Taking Exercises

1. Figure A-4.1(a) shows the exercises form.
2. Click on the list box to select a chapter.
3. Click on the radio button to answer the question.
4. If user answers wrongly, correct answer will show.
5. Click on the "Show Answer" to see the correct answer.
6. User can choose to jump to others question or go to others chapter by clicking on the combo box.

Figure A-4.1(a): Taking Exercises

Chapter 5 Quiz Module

5.1 Taking Quiz According To Chapter

1. In Figure A-5.1(a), users select a chapter and total questions to be answered.
2. Users click on the Proceed button to take the quiz. A pop up message box appears for confirmation.
3. In Figure A-5.1(b), click on the A, B, C or D to start answering.
4. Click on the "Next" or "Previous" button to go to next or previous question.
5. Click on the selection box to go back to other questions.
6. Click on the "Finished" button after finished answering. A user's record is saved in database and the answers are given.

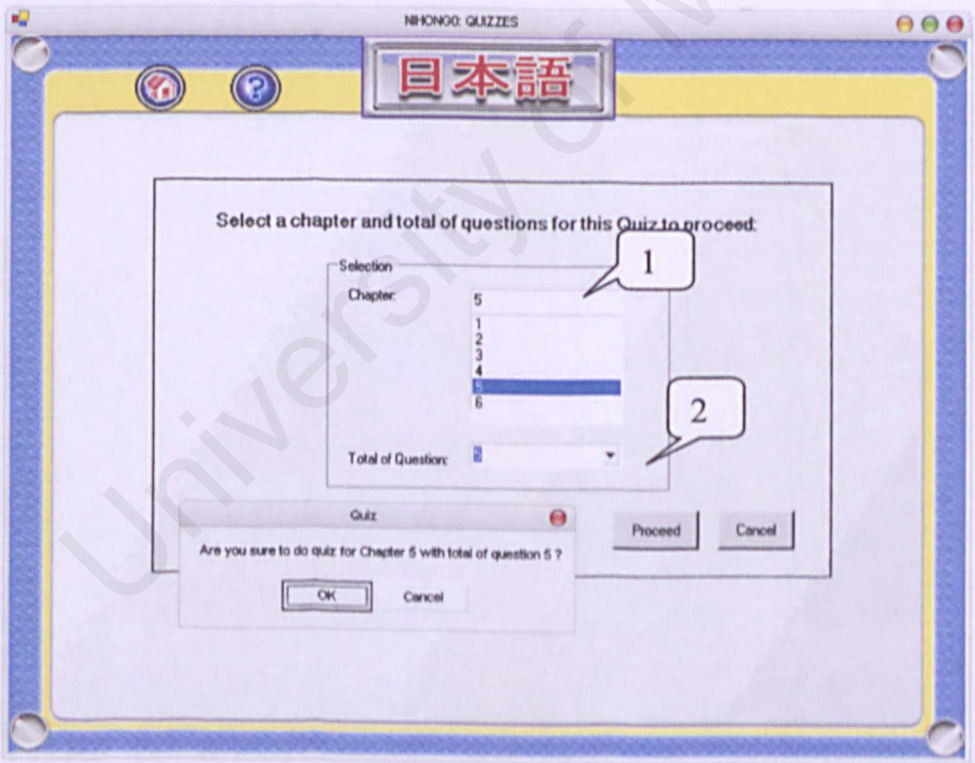

Figure A-5.1(a): Quiz Selection

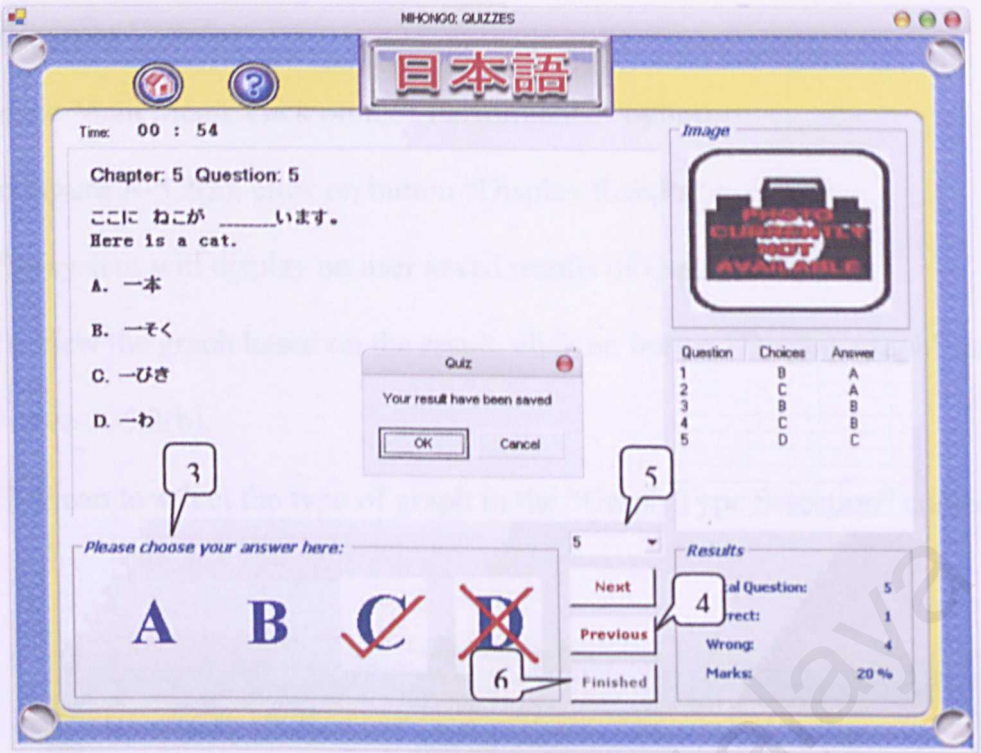

Figure A-5.1(b): Start Taking Quizzes

5.2 Performance Viewing

1. In the Main Menu, click on the "Performance" button.
2. In Figure A-5.2(a), click on button "Display Results".
3. The system will display on user saved results of Quiz module.
4. To view the graph based on the result, click on button "Display graph" as in Figure A-5.2(b).
5. User can to select the type of graph in the "Graph Type Selection" combo box.

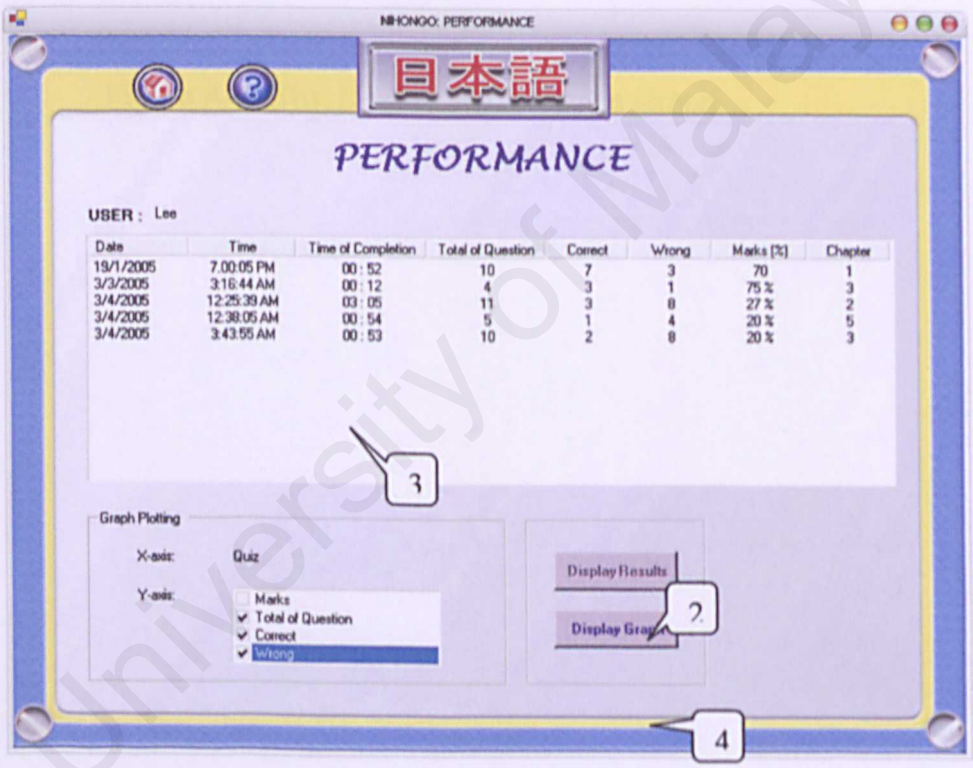

Figure A-5.2(a): Performance Viewing

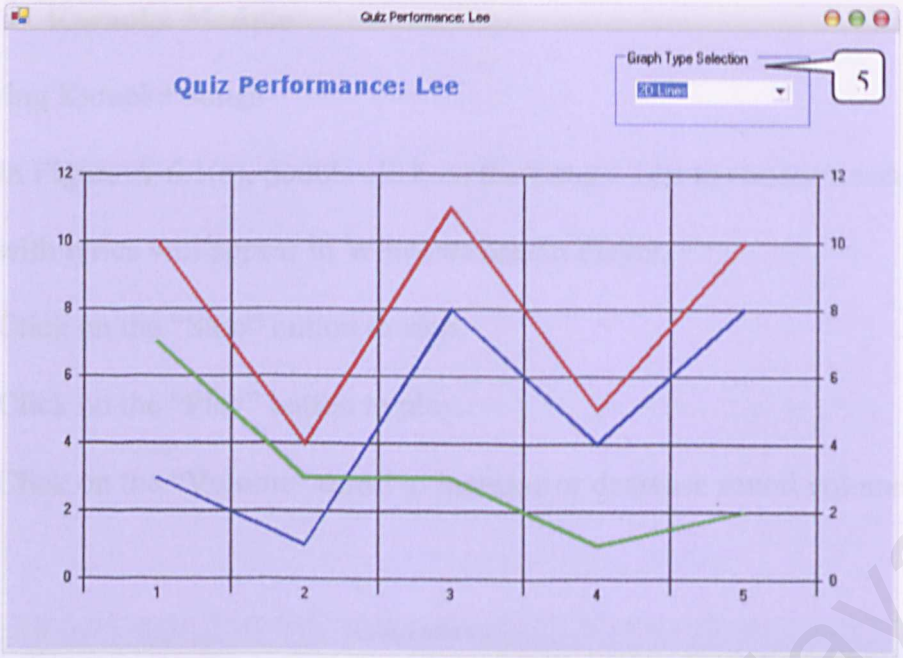

Figure A-5.2(b): Display Performance Graph

Chapter 6 Karaoke Module

6.1 Playing Karaoke Songs

1. In Figure A-6.1(a), double click on the Songs' List to choose a song. Song with lyrics will appear in Windows Media Player.
2. Click on the "Stop" button to stop.
3. Click on the "Play" button to play.
4. Click on the "Volume" scroll to increase or decrease sound volume.

Figure A-6.1(a) Playing Karaoke Songs

Chapter 7 Games Module

7.1 Games: Who Wants To Be A Millionaire

1. Click "Fun" button and then "Games" button from Figure A-1.2(a).
2. In Figure A-7.1(a), click "Millionaire" button to go into the selected game as in Figure A-7.1(b).
3. User can answer the question by selecting one of the four selection buttons.
4. Besides user can call a friend to answer the question.
5. User also can ask the audience to answer the question.
6. User can use the 50-50 chances by answering 1 of the 2 possible answers.
7. User's scored money will be shown at the right hand side column.
8. Once user answer wrongly, he or she will be sent out from the form

Figure A-7.1(a): Game Choices

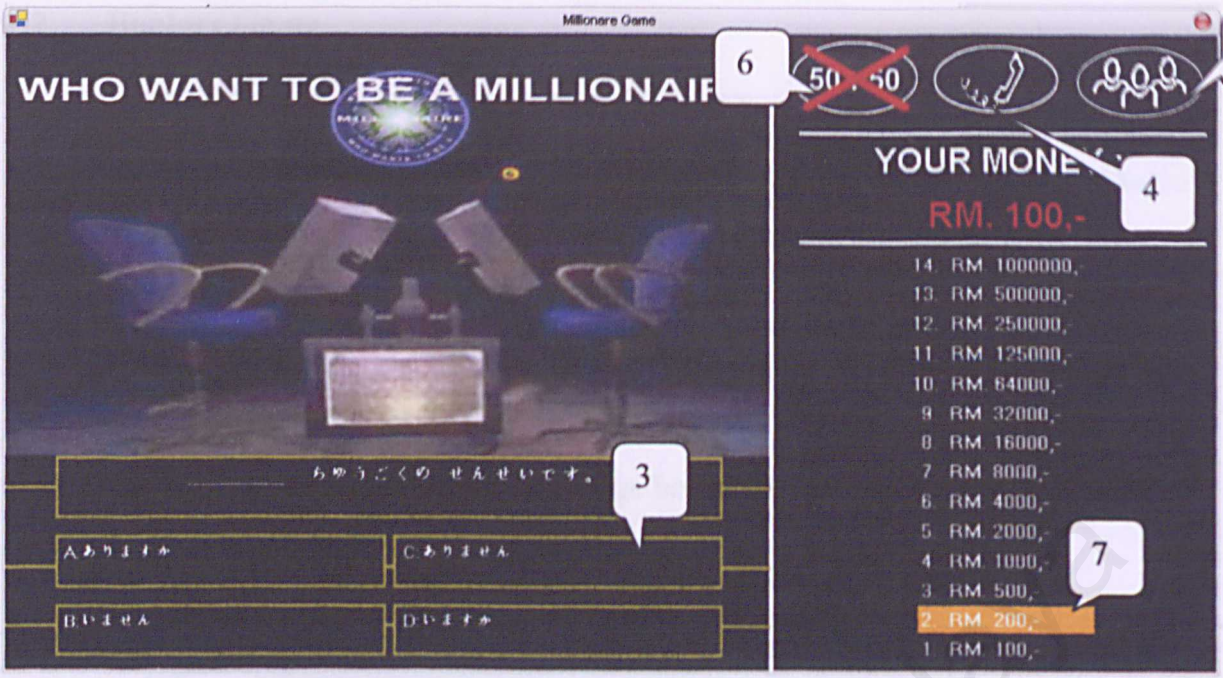

Figure A-7.1(b): Who Want To Be A Millionaire

7.2 Boulder Game

1. The Figure A-7.2(a) shows the Boulder game module.
2. User can select level 1, 2 or 3 and the then click at "Play Game" at the main menu to start.
3. User need to push all the letters together with the correct sequence.
4. Lastly, user can click "CHECK" to check whether their answer is correct.
5. User will be informed by a message box regarding their answer.

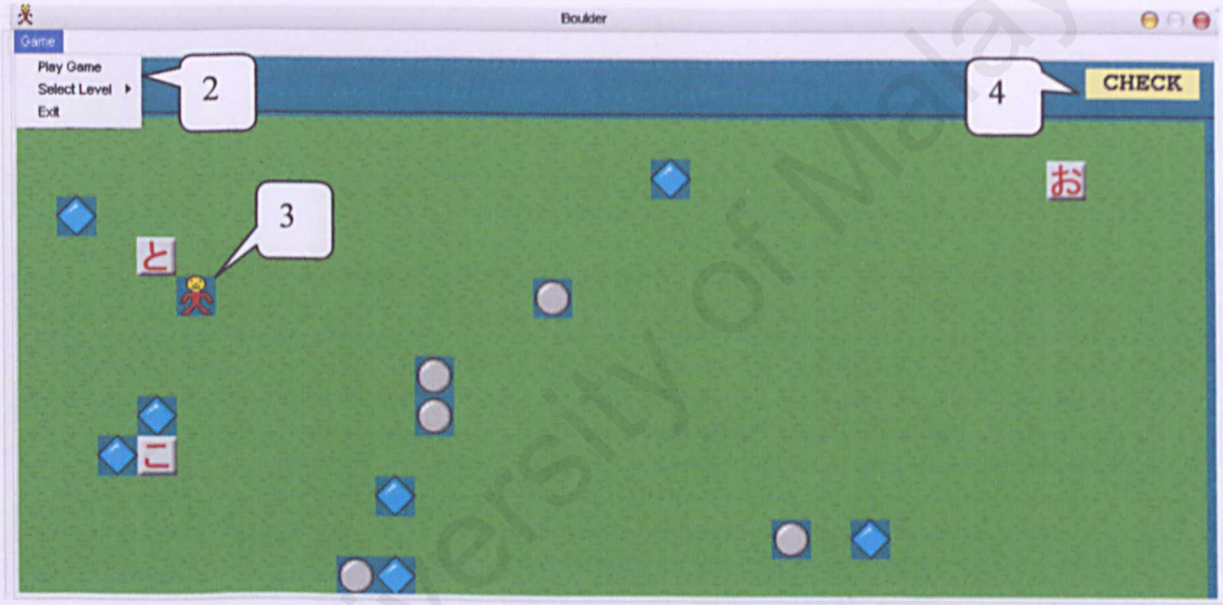

Figure A-7.2(a) Boulder Game

Chapter 8 Maintenance Module

8.1 Administrator's Login

1. Figure A-8.1(a) is the login interface as like user. Only authorize administrator can access the Maintenance
2. Figure A-8.1(b), administrator click on the "Maintenance" button to enter the Main Maintenance.
3. Administrator can click either, Notes, Dictionary, Exercise and Quiz for maintenance.

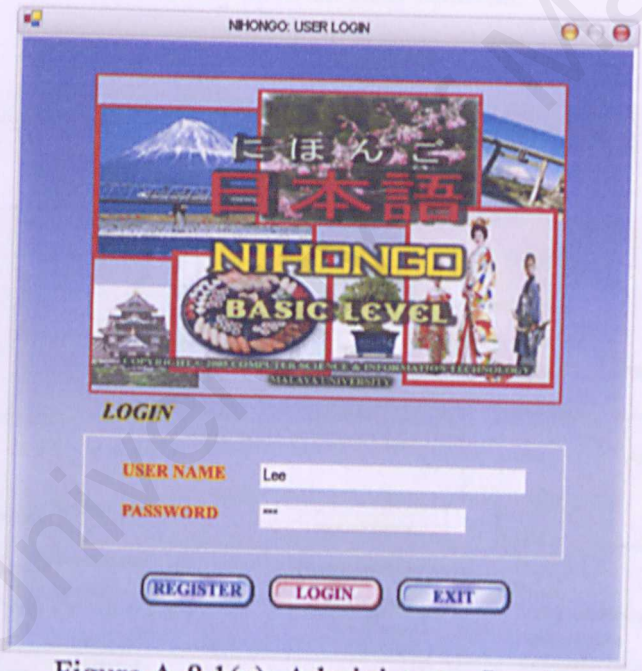

Figure A-8.1(a): Administrator Log In

8.2 Notes Maintenance

8.2.1 Add New Notes

1. Figure A-8.2.1(a) is the Notes Maintenance
2. Click the "New" button to add new records.
3. Fill in the Chapter number. Choose type: Vocabulary or Conversation.
4. If choose Vocabulary, fill in the file name for Vocabulary.
5. If choose Conversation, fill in the data for Conversation.
6. Fill in the Picture's image name after saving the picture into the specified folder named picture.
7. Click the "Save" button to save all new records.

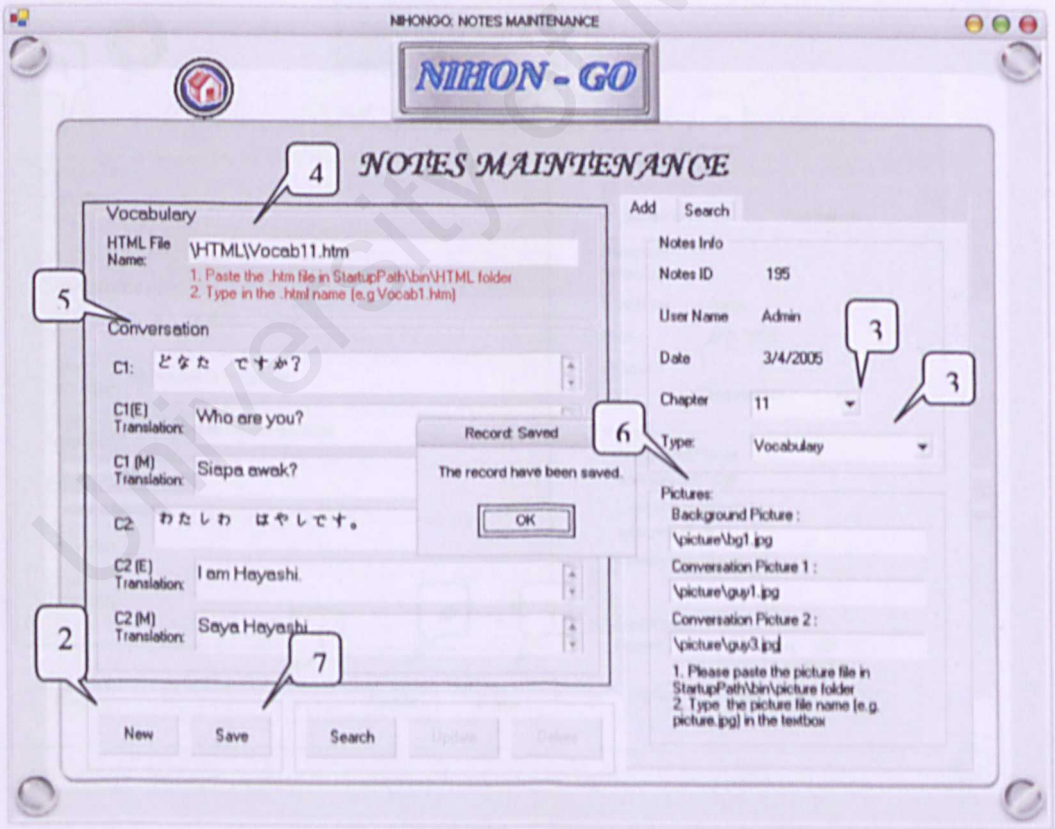

Figure A-8.2.1(a): Notes Maintenance

8.2.2 Search Notes

1. In Figure A-8.2.2(a), click the "Search" button to show the "Search" bar.
2. Fill in the required data wanted below "Search" bar.
3. Once filling in the search data, details are shown on the left side. Administrator can fill in the textboxes.
4. Administrator can click the "Update" button to update the notes.
5. Administrator can click the "Delete" button to delete the notes.
6. Once update, Figure A-8.2.2(b) shows confirmation that the update is successful.
7. Once delete, figure A-8.2.2(c) shows confirmation for data deletion.

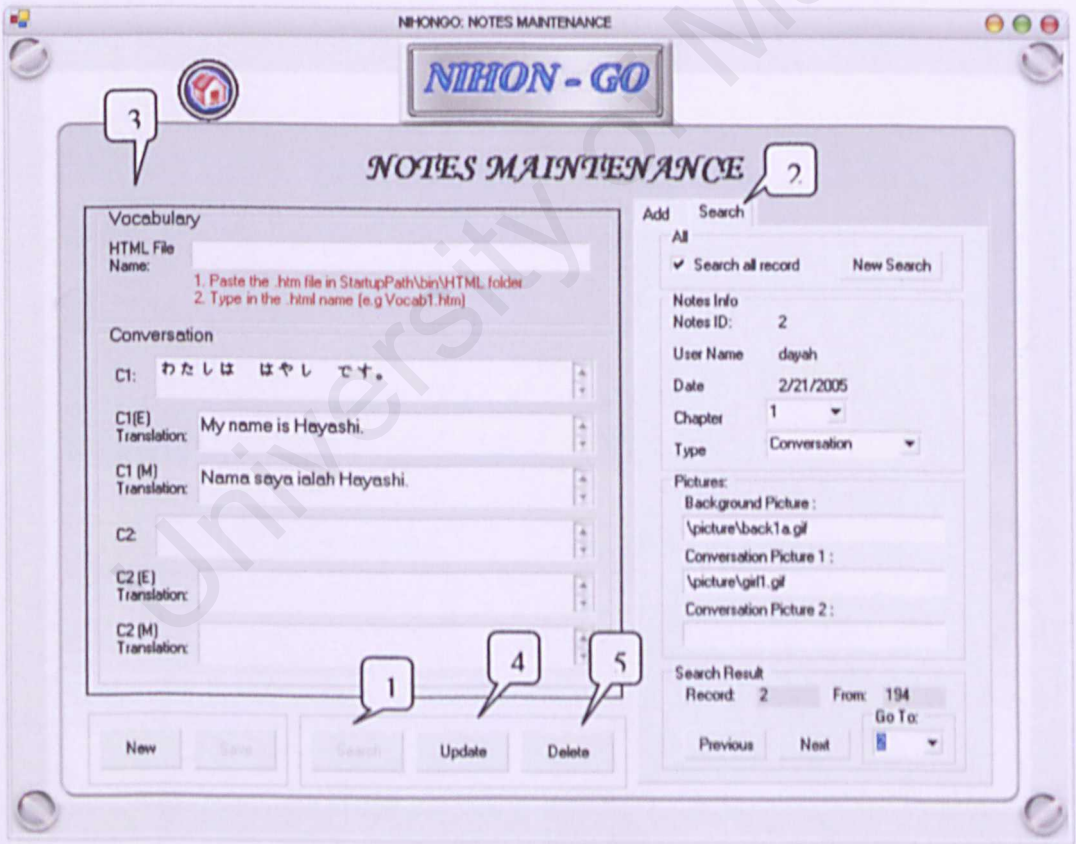

Figure A-8.2.2(a): Notes' Search

Figure A-8.2.2(b): Record Updated

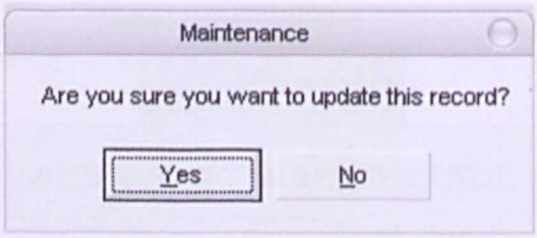

Figure A-8.2.2(c): Data Deletion

8.3 Dictionary Maintenance

8.3.1 Add New Record

1. In Figure A-8.3.1(a), click the "New" button to add new records.
2. Fill in the textboxes and Chapter at the "Add" bar.
3. Click the "Save" button to save new record.

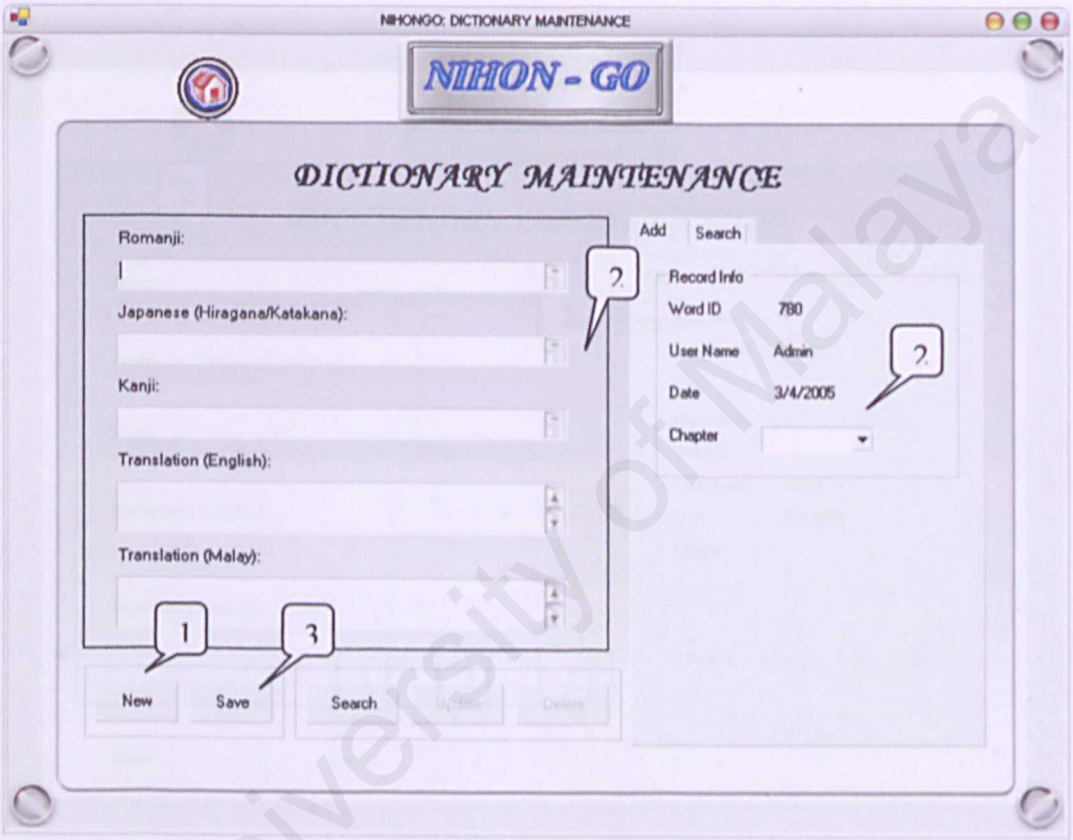

Figure A-8.3.1(a) Add New Record

8.3.2 Search Record In Dictionary

1. In Figure A-8.3.2(a), click the "Search" button to show the "Search" bar.
2. Fill in the Record Info. The data will be shown on the left.
3. Administrator can update or delete the data by clicking "Update" or "Delete" button.

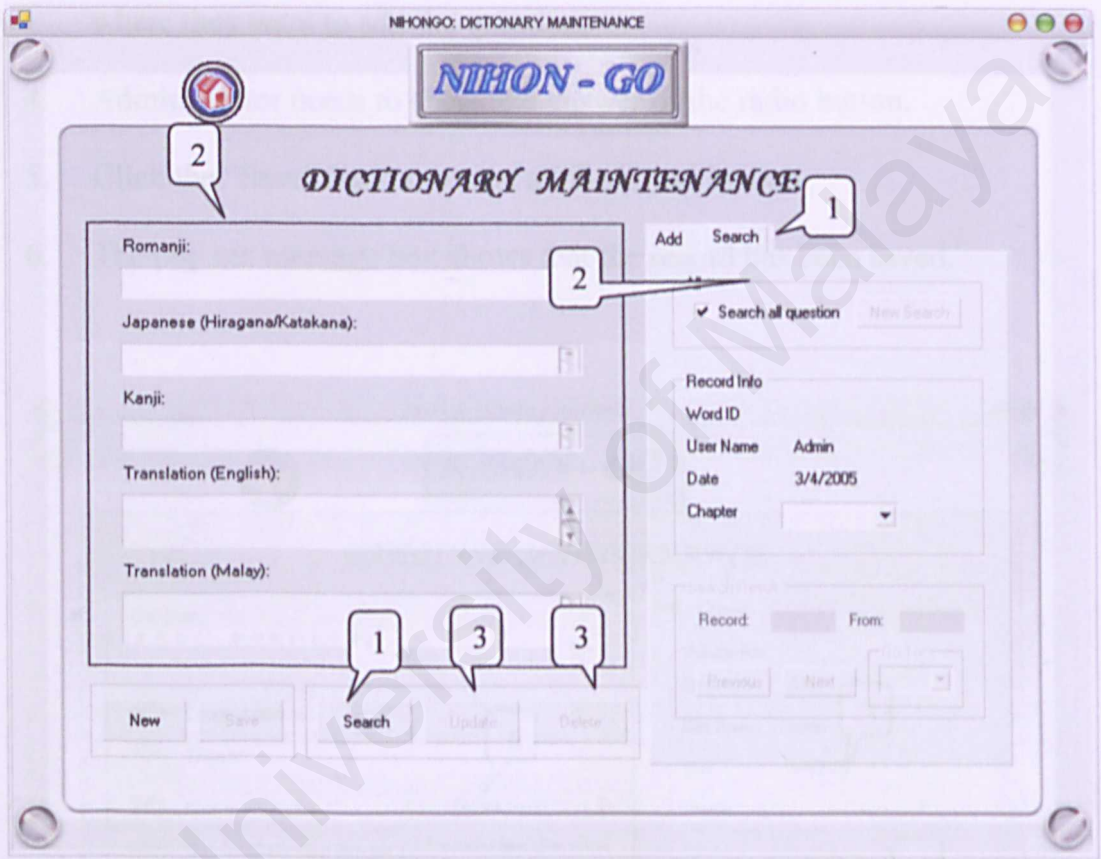

Figure A-8.3.2(a): Search Record

8.4 Exercise Maintenance

8.4.1 Add New Exercise

1. Figure A-8.4.1(a) is the Exercise Maintenance, Administrator click at the "New" button to add the exercises.
2. Administrator needs to fill in all the information in the text boxes.
3. Administrator also can choose "Chapter" from the drop down list to choose where they want to add the question.
4. Administrator needs to check the answer of the radio button.
5. Click the "Save" button to save all the keyed in data.
6. The pop out message box shows that the record has been saved.

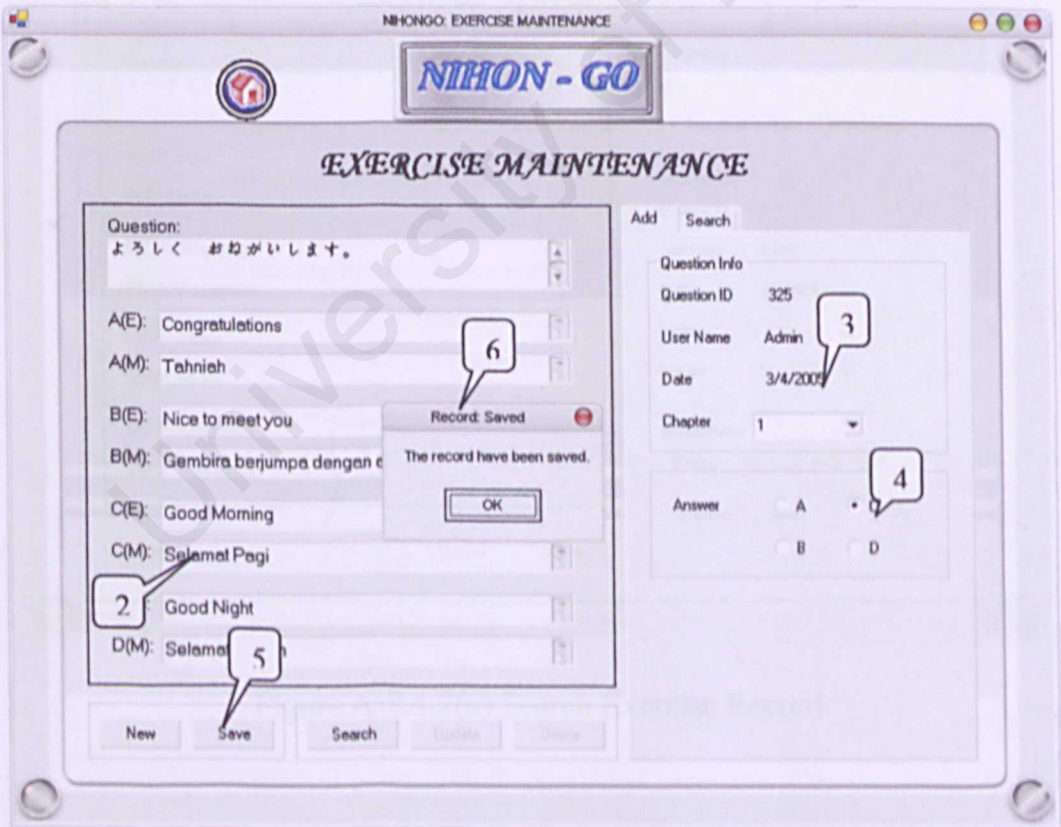

Figure A-8.4.1(a): Add Exercise Record

8.4.2 Search Exercise Record

1. Figure A-8.4.2(a) shows the search record of exercise.
2. Administrator clicks the "Search "button and direct Administrator to the tap strip of tab strip.
3. Administrator needs to check the selected information in the tab strip of "Search".
4. All the relevant information will be displayed regarding the checked information.

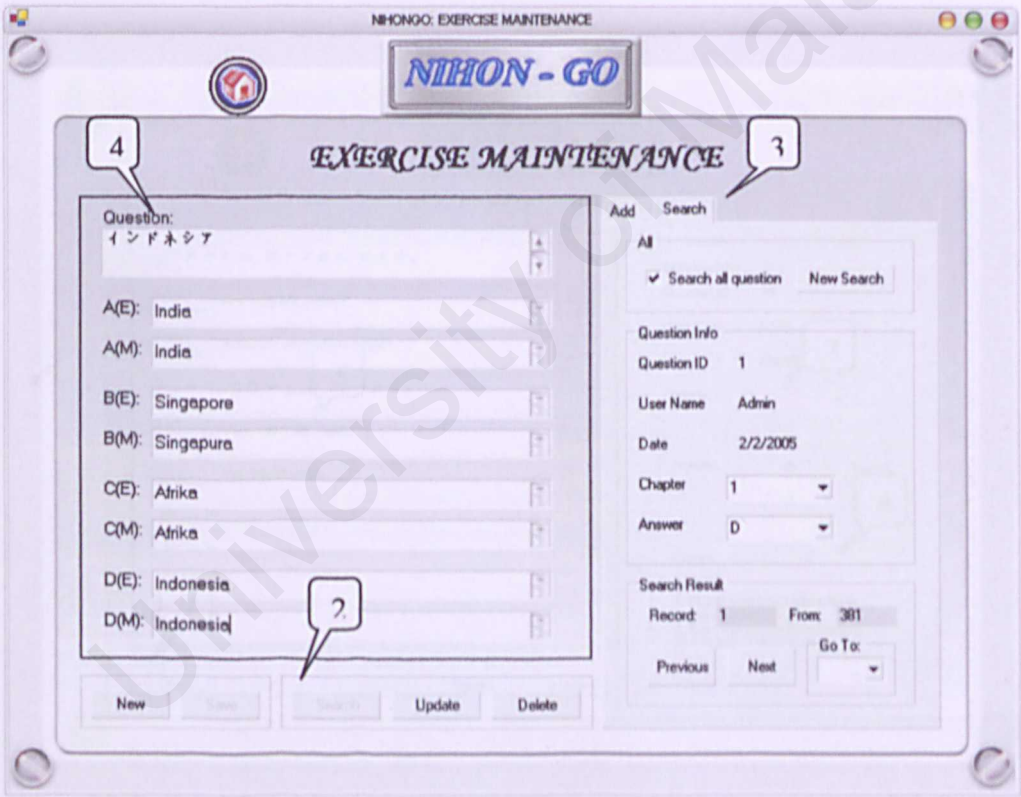

Figure A-8.4.2(a) Search Exercise Record

8.5 Quiz Maintenance

8.5.1 Add new record

1. Figure A-8.5.1(a) is the Quiz Maintenance, Administrator click at the "New" button to add the exercises.
2. Administrator needs to fill in all the information in the text boxes.
3. Administrator also can choose "Chapter" from the drop down list to choose where they want to add the question.
4. Administrator needs to check the answer of the radio button.
5. Click the "Save" button to save all the keyed in data.

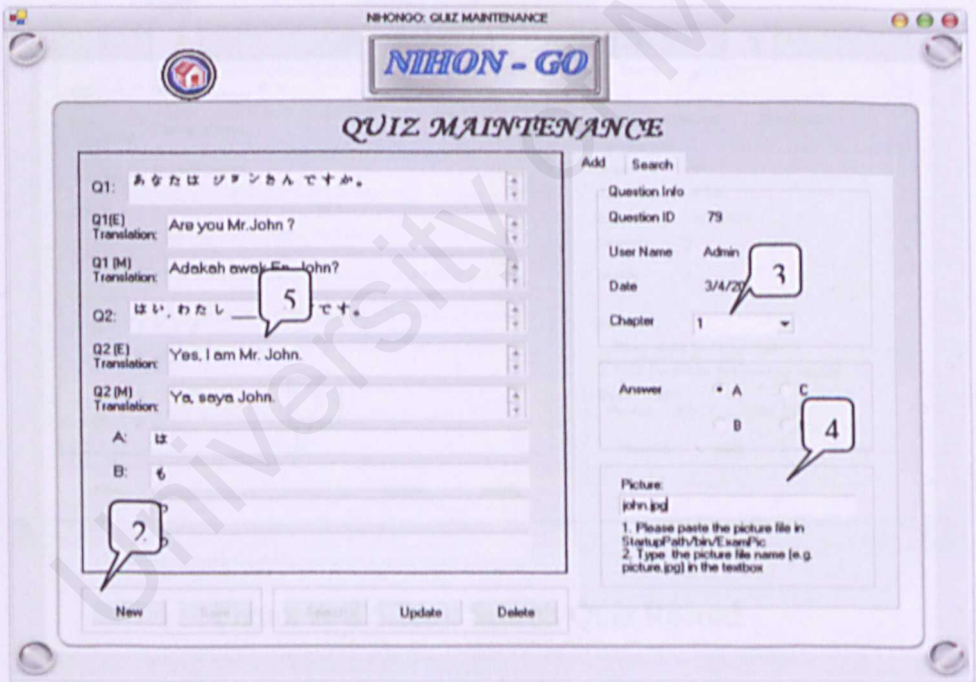

Figure A-8.5.1(a): Add Quiz Record

8.5.2: Search Quiz Record

1. Figure A-8.5.2(a) shows the search record of exercise.
2. Administrator clicks the "Search "button and direct Administrator to the tap strip of tab strip.
3. Administrator needs to check the selected information in the tab strip of "Search".
4. All the relevant information will be displayed regarding the checked information

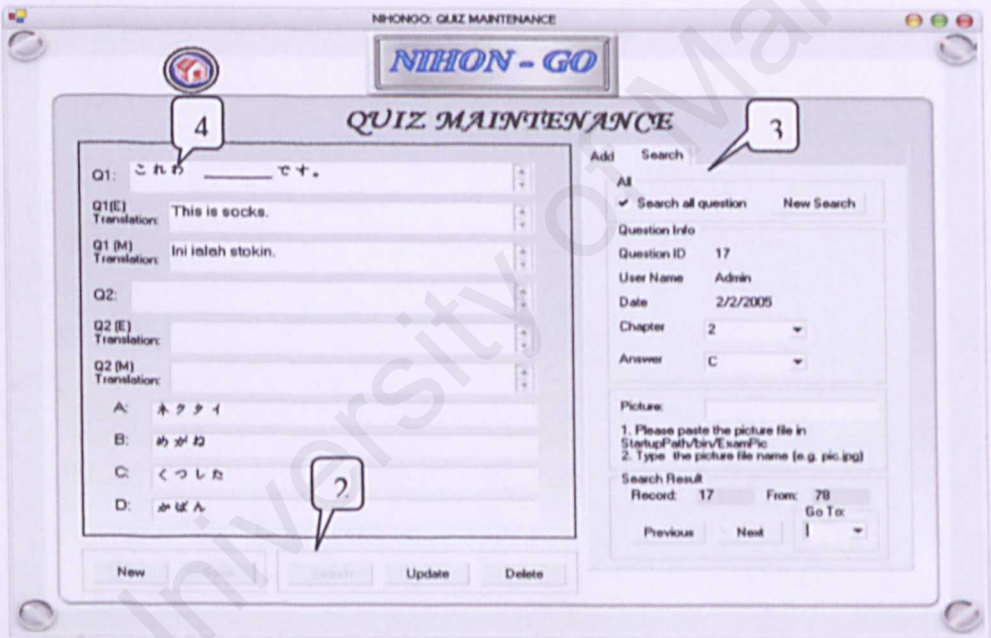

Figure A-8.5.2(a): Search Quiz Record